User's Manual

32

RH850/E1M-S

User's Manual: Hardware

Renesas microcontroller
RH850 Family

Renesas Electronics
www.renesas.com

Rev.1.20 May, 2017

Notice

1. Descriptions of circuits, software and other related information in this document are provided only to illustrate the operation of semiconductor products and application examples. You are fully responsible for the incorporation or any other use of the circuits, software, and information in the design of your product or system. Renesas Electronics disclaims any and all liability for any losses and damages incurred by you or third parties arising from the use of these circuits, software, or information.
2. Renesas Electronics hereby expressly disclaims any warranties against and liability for infringement or any other disputes involving patents, copyrights, or other intellectual property rights of third parties, by or arising from the use of Renesas Electronics products or technical information described in this document, including but not limited to, the product data, drawing, chart, program, algorithm, application examples.
3. No license, express, implied or otherwise, is granted hereby under any patents, copyrights or other intellectual property rights of Renesas Electronics or others.
4. You shall not alter, modify, copy, or otherwise misappropriate any Renesas Electronics product, whether in whole or in part. Renesas Electronics disclaims any and all liability for any losses or damages incurred by you or third parties arising from such alteration, modification, copy or otherwise misappropriation of Renesas Electronics products.
5. Renesas Electronics products are classified according to the following two quality grades: "Standard" and "High Quality". The intended applications for each Renesas Electronics product depends on the product's quality grade, as indicated below.
   "Standard": Computers; office equipment; communications equipment; test and measurement equipment; audio and visual equipment; home electronic appliances; machine tools; personal electronic equipment; and industrial robots etc.
   "High Quality": Transportation equipment (automobiles, trains, ships, etc.); traffic control (traffic lights); large-scale communication equipment; key financial terminal systems; safety control equipment; etc.
   Renesas Electronics products are neither intended nor authorized for use in products or systems that may pose a direct threat to human life or bodily injury (artificial life support devices or systems, surgical implantations etc.), or may cause serious property damages (space and undersea repeaters; nuclear power control systems; aircraft control systems; key plant systems; military equipment; etc.). Renesas Electronics disclaims any and all liability for any damages or losses incurred by you or third parties arising from the use of any Renesas Electronics product for which the product is not intended by Renesas Electronics.
6. When using the Renesas Electronics products, refer to the latest product information (data sheets, user's manuals, application notes, "General Notes for Handling and Using Semiconductor Devices" in the reliability handbook, etc.), and ensure that usage conditions are within the ranges specified by Renesas Electronics with respect to maximum ratings, operating power supply voltage range, heat radiation characteristics, installation, etc. Renesas Electronics disclaims any and all liability for any malfunctions or failure or accident arising out of the use of Renesas Electronics products beyond such specified ranges.
7. Although Renesas Electronics endeavors to improve the quality and reliability of Renesas Electronics products, semiconductor products have specific characteristics such as the occurrence of failure at a certain rate and malfunctions under certain use conditions. Further, Renesas Electronics products are not subject to radiation resistance design. Please ensure to implement safety measures to guard them against the possibility of bodily injury, injury or damage caused by fire, and social damage in the event of failure or malfunction of Renesas Electronics products, such as safety design for hardware and software including but not limited to redundancy, fire control and malfunction prevention, appropriate treatment for aging degradation or any other appropriate measures by your own responsibility as warranty for your products/system. Because the evaluation of microcomputer software alone is very difficult and not practical, please evaluate the safety of the final products or systems manufactured by you.
8. Please contact a Renesas Electronics sales office for details as to environmental matters such as the environmental compatibility of each Renesas Electronics product. Please investigate applicable laws and regulations that regulate the inclusion or use of controlled substances, including without limitation, the EU RoHS Directive carefully and sufficiently and use Renesas Electronics products in compliance with all these applicable laws and regulations. Renesas Electronics disclaims any and all liability for damages or losses occurring as a result of your noncompliance with applicable laws and regulations.
9. Renesas Electronics products and technologies shall not be used for or incorporated into any products or systems whose manufacture, use, or sale is prohibited under any applicable domestic or foreign laws or regulations. You shall not use Renesas Electronics products or technologies for (1) any purpose relating to the development, design, manufacture, use, stockpiling, etc., of weapons of mass destruction, such as nuclear weapons, chemical weapons, or biological weapons, or missiles (including unmanned aerial vehicles (UAVs)) for delivering such weapons, (2) any purpose relating to the development, design, manufacture, or use of conventional weapons, or (3) any other purpose of disturbing international peace and security, and you shall not sell, export, lease, transfer, or release Renesas Electronics products or technologies to any third party whether directly or indirectly with knowledge or reason to know that the third party or any other party will engage in the activities described above. When exporting, selling, transferring, etc., Renesas Electronics products or technologies, you shall comply with any applicable export control laws and regulations promulgated and administered by the governments of the countries asserting jurisdiction over the parties or transactions.
10. Please acknowledge and agree that you shall bear all the losses and damages which are incurred from the misuse or violation of the terms and conditions described in this document, including this notice, and hold Renesas Electronics harmless, if such misuse or violation results from your resale or making Renesas Electronics products available any third party.
11. This document shall not be reprinted, reproduced or duplicated in any form, in whole or in part, without prior written consent of Renesas Electronics.
12. Please contact a Renesas Electronics sales office if you have any questions regarding the information contained in this document or Renesas Electronics products.

(Note 1) "Renesas Electronics" as used in this document means Renesas Electronics Corporation and also includes its majority-owned subsidiaries.

(Note 2) "Renesas Electronics product(s)" means any product developed or manufactured by or for Renesas Electronics.

(Rev.3.0-1 November 2016)

Notes for CMOS devices

(1) Voltage application waveform at input pin:

Waveform distortion due to input noise or a reflected wave may cause malfunction. If the input of the CMOS device stays in the area between VIL (MAX) and VIH (MIN) due to noise, etc., the device may malfunction. Take care to prevent chattering noise from entering the device when the input level is fixed, and also in the transition period when the input level passes through the area between VIL (MAX) and VIH (MIN).

(2) Handling of unused input pins:

Unconnected CMOS device inputs can be cause of malfunction. If an input pin is unconnected, it is possible that an internal input level may be generated due to noise, etc., causing malfunction. CMOS devices behave differently than Bipolar or NMOS devices. Input levels of CMOS devices must be fixed high or low by using pull-up or pull-down circuitry. Each unused pin should be connected to power supply or GND via a resistor if there is a possibility that it will be an output pin. All handling related to unused pins must be judged separately for each device and according to related specifications governing the device.

(3) Precaution against ESD:

A strong electric field, when exposed to a MOS device, can cause destruction of the gate oxide and ultimately degrade the device operation. Steps must be taken to stop generation of static electricity as much as possible, and quickly dissipate it when it has occurred. Environmental control must be adequate. When it is dry, a humidifier should be used. It is recommended to avoid using insulators that easily build up static electricity. Semiconductor devices must be stored and transported in an anti-static container, static shielding bag or conductive material. All test and measurement tools including work benches and floors should be grounded. The operator should be grounded using a wrist strap. Semiconductor devices must not be touched with bare hands. Similar precautions need to be taken for PW boards with mounted semiconductor devices.

(4) Status before initialization:

Power-on does not necessarily define the initial status of a MOS device. Immediately after the power source is turned ON, devices with reset functions have not yet been initialized. Hence, power-on does not guarantee output pin levels, I/O settings or contents of registers. A device is not initialized until the reset signal is received. A reset operation must be executed immediately after power-on for devices with reset functions.

(5) Power ON/OFF sequence:

In the case of a device that uses different power supplies for the internal operation and external interface, as a rule, switch on the external power supply after switching on the internal power supply. When switching the power supply off, as a rule, switch off the external power supply and then the internal power supply. Use of the reverse power on/off sequences may result in the application of an overvoltage to the internal elements of the device, causing malfunction and degradation of internal elements due to the passage of an abnormal current. The correct power on/off sequence must be judged separately for each device and according to related specifications governing the device.

(6) Input of signal during power off state:

Do not input signals or an I/O pull-up power supply while the device is not powered. The current injection that results from input of such a signal or I/O pull-up power supply may cause malfunction and the abnormal current that passes in the device at this time may cause degradation of internal elements. Input of signals during the power off state must be judged separately for each device and according to related specifications governing the device.

How to Use This Manual

1. Purpose and Target Readers

This manual is designed to provide the user with an understanding of the hardware functions and electrical characteristics of the MCU. It is intended for users designing application systems incorporating the MCU. A basic knowledge of electric circuits, logical circuits, and MCUs is necessary in order to use this manual.
The manual comprises an overview of the product; descriptions of the CPU, system control functions, peripheral functions, and electrical characteristics; and usage notes.

Particular attention should be paid to the precautionary notes when using the manual. These notes occur within the body of the text, at the end of each section, and in the Usage Notes section.

The revision history summarizes the locations of revisions and additions. It does not list all revisions. Refer to the text of the manual for details.

The following documents apply to the RH850/E1x Group. Make sure to refer to the latest versions of these documents. The newest versions of the documents listed may be obtained from the Renesas Electronics Web site.

Document Type	Description	Document Title	Document No.
Data sheet	Overview of hardware and electrical characteristics		
User's manual: Hardware	Hardware specifications (pin assignments, memory maps, peripheral function specifications, electrical characteristics, timing charts) and operation description Note: Refer to the application notes for details on using peripheral functions.	RH850/E1M-S User's manual: Hardware	This User's Manual
User's manual: Software	Description of CPU instruction set	RH850G3M User's Manual: Software	R01US0123EJ0100
		RH850G3K User's Manual: Software	R01US0125EJ0100
Application note	Information on using peripheral functions and application examples Sample programs Information on writing programs in assembly language and C		
Renesas technical update	Product specifications, updates on documents, etc.		

2. Notation of Numbers and Symbols

The notation conventions for register names, bit names, numbers, and symbols used in this manual are described below.

(1) Register Names, Bit Names, and Pin Names

Registers, bits, and pins are referred to in the text by symbols. The symbol is accompanied by the word “register,” “bit,” or “pin” to distinguish the three categories.

Examples: the DEF bit in the ABC register
PA01 pin, VCC pin

(2) Notation of Numbers

Binary numbers are notated as nnnnB (However, the “B” may be omitted when it is clear that the number is binary), hexadecimal numbers are notated as nnnnH, and decimal numbers are notated as nnnn.

Examples Binary: 11B or 11
Hexadecimal: EFA0H
Decimal: 1234

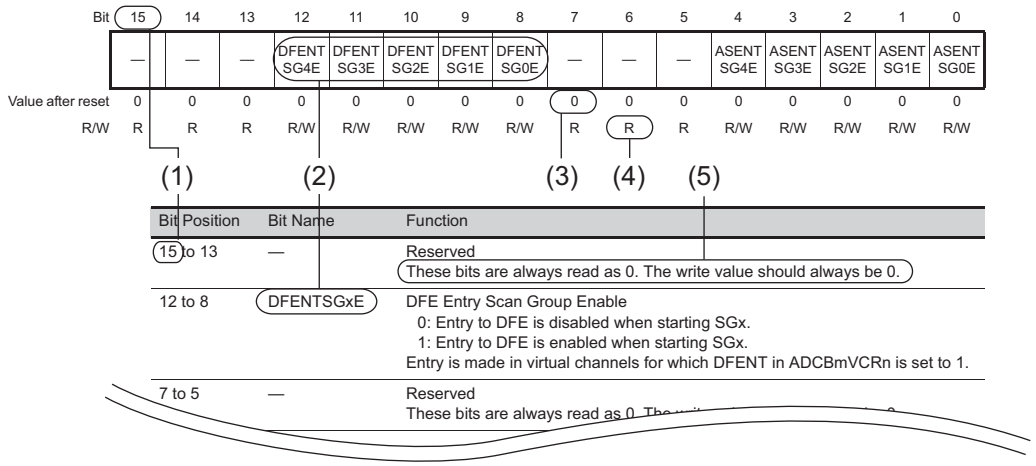
(3) Notation for Active-Low Signals

An overbar is added to the names of active-low signals.

Example $\overline{\text{ABCDEF}}$

3. Register Notation

Each section in this manual provides a table listing all the registers used by the corresponding module before the register descriptions in the section. The symbols and terms used in these tables are described below.



(1) Bit Position

Indicates the bit number.

The bits are shown in the order 31 to 0 for 32-bit registers, in the order 15 to 0 for 16-bit registers, and in the order 7 to 0 for 8-bit registers.

(2) Bit Name

The short form of the name of the bit or bit field within the register.

When the individual bits of bit fields have to be clearly indicated, notation allowing this is included (e.g. ABCD[3:0]).

A reserved bit is indicated by "—".

Instead of a bit name, a blank is used for some bits, such as those of timer counters.

(3) Value after reset

Indicates the value of each bit after a reset, i.e., the value after reset.

0: Value after reset is 0

1: Value after reset is 1

—: Value after reset is undefined

(4) R/W

Indicates whether each bit is readable or writable, or either writing to or reading from the bit is prohibited.

R/W: Bit or field is readable and writable.

R/(W): Bit or field is readable or writable.
However, writing is only performed to clear the flag.

R: Bit or field is readable.
However, "R" is indicated for all reserved bits. When writing to the bit is required, write the value stated in the bit table or the initial value.
In case of writing to writable registers that also include non-reserved bits with the R-attribute, writing to the R-attribute bits will be ignored unless otherwise specified.

W: Bit or field is writable.
However, only the value in the bit table is guaranteed when reading from the bit.

(5) Function

Describes the function enabled by setting the bit.

Table of Contents

Section 1 Overview

The RH850/E1M-S is a product of the single-chip microcomputer RH850 series of Renesas Electronics.

This section describes an overview of the RH850/E1M-S.

1.1 Outline

This product is a 32-bit single-chip microcomputer that incorporates multiple CPUs of the RH850 Series, code flash, RAM, DMA controller, CAN, high-speed communication interfaces including RHSB (microsecond bus), two types (sequential conversion type and delta-sigma type) of A/D converters, peripheral functions including digital filter processing, and an advance timer unit (ATU-IV) that is best suited for high-speed accurate power train control. This product also conforms to the Automotive Safety Integrity Level (ASIL) that is highly demanded in the recent automotive field.

(1) Installing multiple RH850 cores

This product contains a single RH850G3M as the main CPU (referred to as CPU1 hereafter) and a single RH850G3K as the peripheral control CPU (referred to as PCU: Peripheral Control Unit hereafter). The CPU1 and the PCU supports RISC-type instruction sets and have significantly improved the instruction execution speed with basic instructions (one clock cycle per instruction) and the optimized 7- and 5-stage pipeline configurations. Furthermore, this product also supports multiplication instructions using a 32-bit hardware multiplier, saturated product-sum operation instructions, and bit manipulation instructions as instructions best suited for various fields such as applications of the automobile power train control.

Two-byte basic instructions and high-level language instructions improve object code efficiency for the C compiler and reduce the program size. Furthermore, this product is suited for advanced real-time control fields thanks to high-speed interrupt response time including the processing time of the on-chip interrupt controller.

(2) On-chip code flash and data flash

This product incorporates 4-Mbyte code flash allowing high-speed accesses, which enables each CPU to access this flash memory. This memory can be reprogrammed while it is placed on an application system. This can shorten the system development period and significantly improve the serviceability after the system is delivered.

This product also has 64-Kbyte data flash that is available for storing EEPROM data.

(3) A variety of peripheral functions

The RH850/E1M-S incorporates timers (ATU-IV and APA) best suited for reducing hardware and software control loads in the automobile power train control (e.g. engine, transmission), digital filter engine (DFE), two types of A/D converters (ADCB and DS-ADC), and other devices. In addition, the RH850/E1M-S incorporates standard peripheral functions (UART, CAN, and RHSB) for the automotive field. It also has the global standard on-chip debug interface (NEXUS JTAG) as a debug interface. This allows construction of systems without providing these functions externally, reducing costs, quantity of components, and footprint.

(4) Conforming to functional safety

The memory protection function, peripheral device protection function, system/register protection function, and timing monitoring function are provided to ensure highly reliable operations of the CPU. With these function, the RH850/E1M-S conforms to the functional safety requirements (ASIL) in the automotive field.

1.2 Features

CPU1 core	RH850G3M: 1 unit (for high-speed operation and control)
CPU1 cache memory	8 Kbytes
Minimum CPU1 instruction execution time	3.125 ns (during internal 320 MHz operation)
General CPU1 registers	Thirty-two 32-bit registers
CPU1 instruction sets	Signed multiplication (32 bits × 32 bits → 64 bits): 1 to 2 CPU clocks Saturated operation instructions (with overflow/underflow detection function) 32-bit arithmetic/logical shift instructions: 1 CPU clock Bit manipulation instructions Load/store instructions with long/short formats Signed load instructions
PCU core	RH850G3K: 1 unit (for medium-speed operation and peripheral control)
Minimum PCU instruction execution time	12.5 ns (during internal 80 MHz operation)
General PCU registers	Thirty-two 32-bit registers
PCU instruction sets	Signed multiplication (32 bits × 32 bits → 64 bits): 1 and 2 CPU clocks Saturated operation instructions (with overflow/underflow detection function) 32-bit arithmetic/logical shift instructions: 1 CPU clock Bit manipulation instructions Load/store instructions with long/short formats Signed load instructions
Memory space	4-Gbyte address space (common to program and data) (128 Mbytes accessible by the PCU)
Code flash	Two types of memory area • User area: 4 Mbytes (common to CPU1 and PCU) • User boot area: 32 Kbytes High-speed reading through cache enabled (only for CPU1)
Data flash	64 Kbytes (common to CPU1 and PCU)
RAM	RAM local RAM: 64 Kbytes (CPU1), 32 Kbytes (PCU) Global RAM: 256 Kbytes (common to CPU1 and PCU) (32 Kbytes: Standby RAM) Emulation RAM 8 Kbytes (common to CPU1 and PCU)
Interrupts/exceptions	1 nonmaskable interrupt (NMI pin) 1 FE level interrupt (ECM) 483maskable interrupts (high-speed: 21, low-speed: 462) Simultaneous distribution of interrupt sources to multiple cores (CPU1, PCU) • Applicable sources: non-maskable interrupt (NMI pin), FE level interrupt, 21 high-speed maskable interrupts External interrupt input function (IRQ pins) Software interrupt function (SINT) Inter-processor interrupt function (IPIR) 16-level priority specifiable for maskable interrupts For RH850 G3M and G3K exceptions, see “RH850G3M User's Manual: Software”, and “RH850G3K User's Manual: Software”.
DMA controller	16 channels incorporated (8 channels × 2 modules) Transfer unit: 8 bits/16 bits/32 bits/64 bits/128 bits Dual-address transfer mode Address reloading function Chain transfer function Three transfer modes settable: Single transfer, block transfer 1 (specified by number of transfer times), and block transfer 2 (specified by address count) Transfer target: On-chip memory, on-chip peripheral modules (excluding DMAC and DTS) Transfer requests can be set by interrupt sources and the software.

DTS controller	128 channels incorporated Transfer unit: 8 bits/16 bits/32 bits/64 bits/128 bits Dual-address transfer mode Address reloading function Chain transfer function Three transfer modes settable: Single transfer, block transfer 1 (specified by number of transfer times), and block transfer 2 (specified by address count) Transfer target: On-chip memory, on-chip peripheral modules (excluding DMAC and DTS) Transfer requests can be set by interrupt sources and the software.
I/O	Output driving ability of specific input/output pins selectable Inversion or non-inversion of output values of specific input/output pins selectable Pull-down or pull-down off of specific input/output pins selectable
Safety functions	Flash memory ECC error detection function RAM ECC error detection function Peripheral module RAM ECC error detection function (DTS, APA, RS-CAN, FlexRay, DFE) Oscillation stop detection function and oscillation function by the backup clock Error control module (ECM) Duplexing of modules (e.g. CPU1, ECM, error output pins)
Error control module (ECM)	Collects information of each error check system and safety function and indicates error status. When an error is detected, an error signal can be output from the error pin to the external. Interrupts and internal reset signals can be generated upon detection of an error. Provided with a function to generate a pseudo-error for debugging and self-diagnosis.
Data CRC function	The data CRC (cyclic redundancy check) function can verify or generate data streams protected by CRC with various lengths and different bit widths.
Multi-Input signature generator (MISG)	Monitors write access to specific addresses by the CPU, and generates a 64-bit signature using the write data.
Window watchdog timer (WDTA)	Two channels incorporated Can generate a signal to the ECM when an error occurs. Can generate an interrupt at 75% of the error source (register write during counter overflow or window closing period) window.
Advanced timer unit IV (ATU-IV)	Timer A: 32-bit input capture × 7 channels Timer B: Angle clock generating timer × 1 channel Timer C: 32-bit input capture/output compare × 32 channels Timer D: 32-bit one-shot pulse × 40 channels 32-bit input capture × 40 channels 32-bit output compare × 40 channels Timer E: 24-bit PWM x 28 channels Timer F: 32-bit event counter × 16 channels Timer G: 32- bit interval timer × 10 channels Timer H: 32-bit interval timer × 1 channel Timer J: 32-bit input capture (with 9-stage FIFO) × 6 channels
Autonomous pulse adapter (APA)	16 channels incorporated Changes output at an event (compare match) with no CPU load using values of peripheral modules (ATU-IV, ADCB, DS-ADC) as reference. Can generate output patterns automatically by combining multiple events.
Motor control timer (TSG2) Timer option (TAPA)	TSG2, TAPA: Two units are incorporated individually. • Incorporates a timer unit that can control up to two 3-phase motor controls (U, V, and W). • Can forcibly set the TSG2 timer output to Hi-Z when an error is detected.
OS timer (OSTM)	Three units incorporated • A 32-bit timer assuming the use of OS • Interval timer mode or free-running timer mode settable • Synchronous start between units settable
Peripheral interconnection function (PIC)	One unit incorporated • Synchronous operation using TSG2 timer and timer input/output can be connected between timers.

Serial communication interface 3 (SCI3)	Four channels incorporated • Clock synchronization or start-stop system settable • Full-duplex communication enabled • Arbitrary bit rate settable by the on-chip baud rate generator • LSB first or MSB first selectable
Serial interface H (CSIH)	Four channels incorporated • Three-wire serial synchronous data transfer • Master mode or slave mode selectable • Settable six or four chip select output signals • Arbitrary bit rate settable by the on-chip baud rate generator
CAN interface (RS-CAN)	Four channels incorporated • Conforming to ISO-11898 • Maximum transfer speed: 1 Mbps • A total of 320 message buffers provided for 4 channels • On-chip reception filtering function
FlexRay	Two channels incorporated (Ach, Bch) • Conforming to Protocol Specification 2.1 • Buffer size: A 8-Kbyte space is divided into up to 128 (for transmission, reception, and receive FIFO) • Message filtering: ID filter, channel filter, cycle counter filter • Bit rate: 10 Mbps
LIN master interface (RLIN2)	One channel incorporated • Settable to support various frame sizes • Checksum type settable • Wake-up transmission/reception enabled • Arbitrary bit rate (2400, 9600, 10417, 19200 bps) settable by the on-chip baud rate generator
Renesas high-speed bus (RHSB)	Two channels incorporated Communication module supporting microsecond bus channels • Down-stream communication • Up to two slaves settable individually • Functions for emergencies
AD converter (ADCB)	48 channels incorporated • 12-bit resolution, sequential conversion • Conversion speed: 1.0 µs • Independent operation of 12-bit resolution for 24 channels (ADC0) and 12-bit resolution for 24 channels (ADC1) • Track and hold circuit (ADC0: 2 channels, ADC1: 2 channels) • Scan groups for five systems settable • Two scan modes (multi-scan mode and continuous scan mode) • Up to 40 virtual channels settable • Two types of A/D conversion and addition functions incorporated • Converts the converted results to floating-point format. • Can enter data directly to the digital filter engine. • Safety functions (including self-diagnosis, data register upper limit/lower limit check, and data register parity)
Delta-sigma AD converter (DS-ADC)	16 channels incorporated (8 channels × 2 inputs) • Advanced A/D conversion • A/D conversion system: Delta-sigma modulation system • Supports single-end input and differential input. • On-chip programmable gain amplifier (x1, x2, x4, and x8) • Can enter data directly to the digital filter engine. • Safety functions (including self-diagnosis, data register upper limit/lower limit check, and data register parity)

Power supply voltage monitor	Monitors VDD voltage internally. If VDD rises above or falls below the specified voltage, an error is reported outside the LSI. Can set a time filter for error detection.
Digital filter	16-channel digital filtering (FIR/IIR) functions • FIR with up to 32 taps selectable • Up to 6-order IIR selectable • Filter coefficients and data are stored in the RAM to be connected. • For filtered data: Accumulation processing or decimation processing enabled Peak-hold processing or comparison processing enabled • Directly inputs converted data from the on-chip A/D converter or delta-sigma AD converter to perform automatic filtering.
On-chip debugging unit (OCD)	NEXUS JTAG: One channel incorporated LDU: One channel incorporated
Boundary scan	Boundary scan compliant with the IEEE 1149.1 standard
Clock controller	An oscillation circuit incorporated 20 MHz oscillation enabled without using external capacitors 20 MHz (maximum) clock can be input from the EXTAL pin. Multiplication of clock enabled by the on-chip PLL A ring oscillator circuit incorporated SSCG for reducing radiation noise incorporated. However, clock without SSCG is supplied to the timer and the communication module.
Operating modes	Operating modes • User boot mode • Serial programming mode Power-down mode • Power shutoff standby
Package	304-pin plastic FBGA (0.8 mm ball pitch) (19 × 19) 252-pin plastic FBGA (0.8 mm ball pitch) (17 × 17)

1.3 Application Fields

- Automotive field (including engine control system and transmission control system)

1.4 Order Information

Product Name	Package	On-Chip ROM	Operating Temperature (T_j)	External Oscillator	Maximum Operating Frequency
R7F701202EABA	Plastic FBGA-304 0.8-mm ball pitch 19 mm × 19 mm	4 MB	max. 150°C	20 MHz	320 MHz
R7F701202EABG	Plastic FBGA-252 0.8-mm ball pitch 17 mm × 17 mm	4 MB	max. 150°C	20 MHz	320 MHz
R7F701204EABA	Plastic FBGA-304 0.8-mm ball pitch 19 mm × 19 mm	4 MB	max. 150°C	20 MHz	240 MHz
R7F701204EABG	Plastic FBGA-252 0.8-mm ball pitch 17 mm × 17 mm	4 MB	max. 150°C	20 MHz	240 MHz

We hereafter refer to FBGA as BGA unless the complete abbreviation is especially required.

1.5 Pin Connection Diagram (Top View)

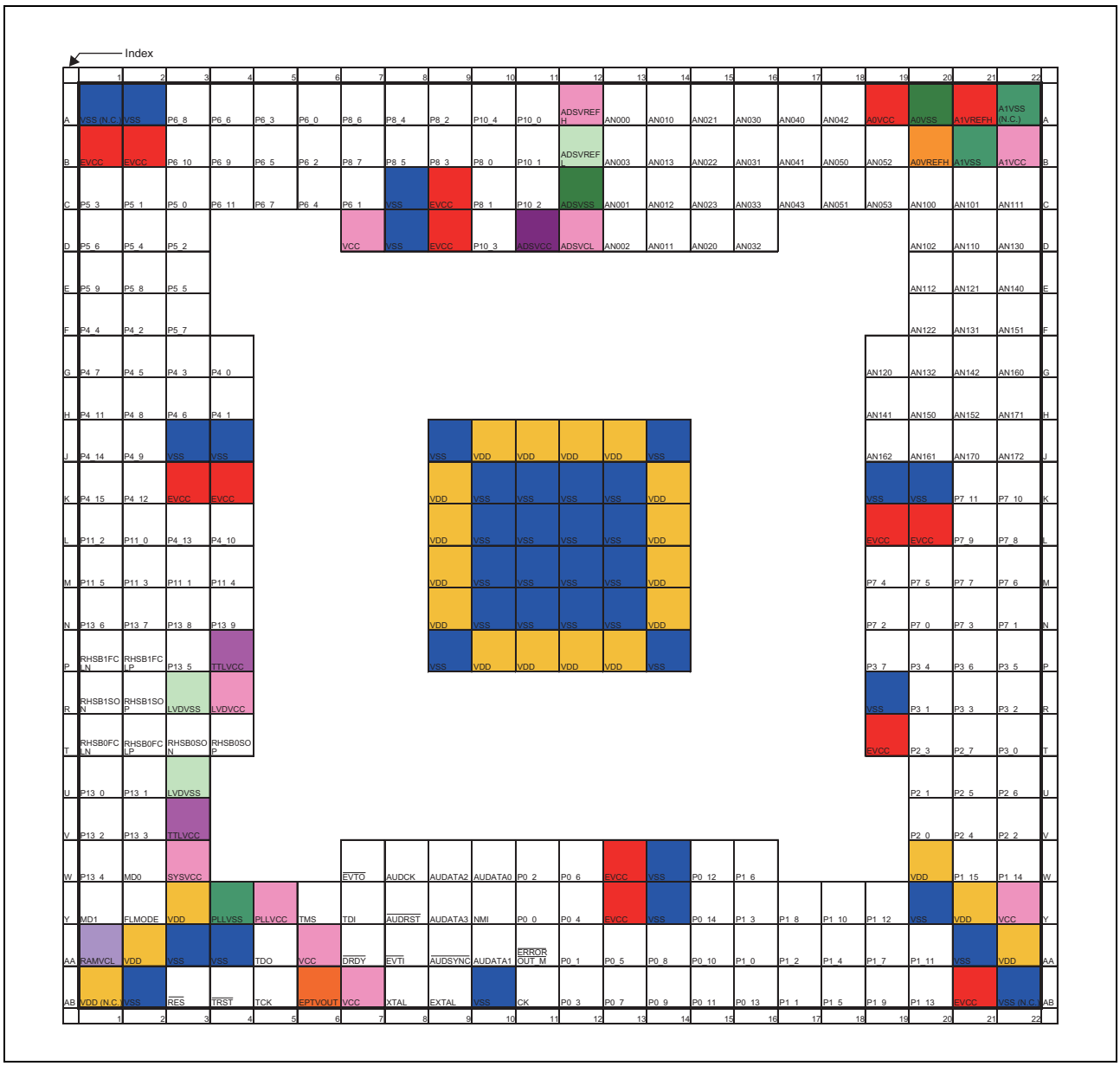


Figure 1.1 Pin Connection Diagram (E1MS-BGA304)

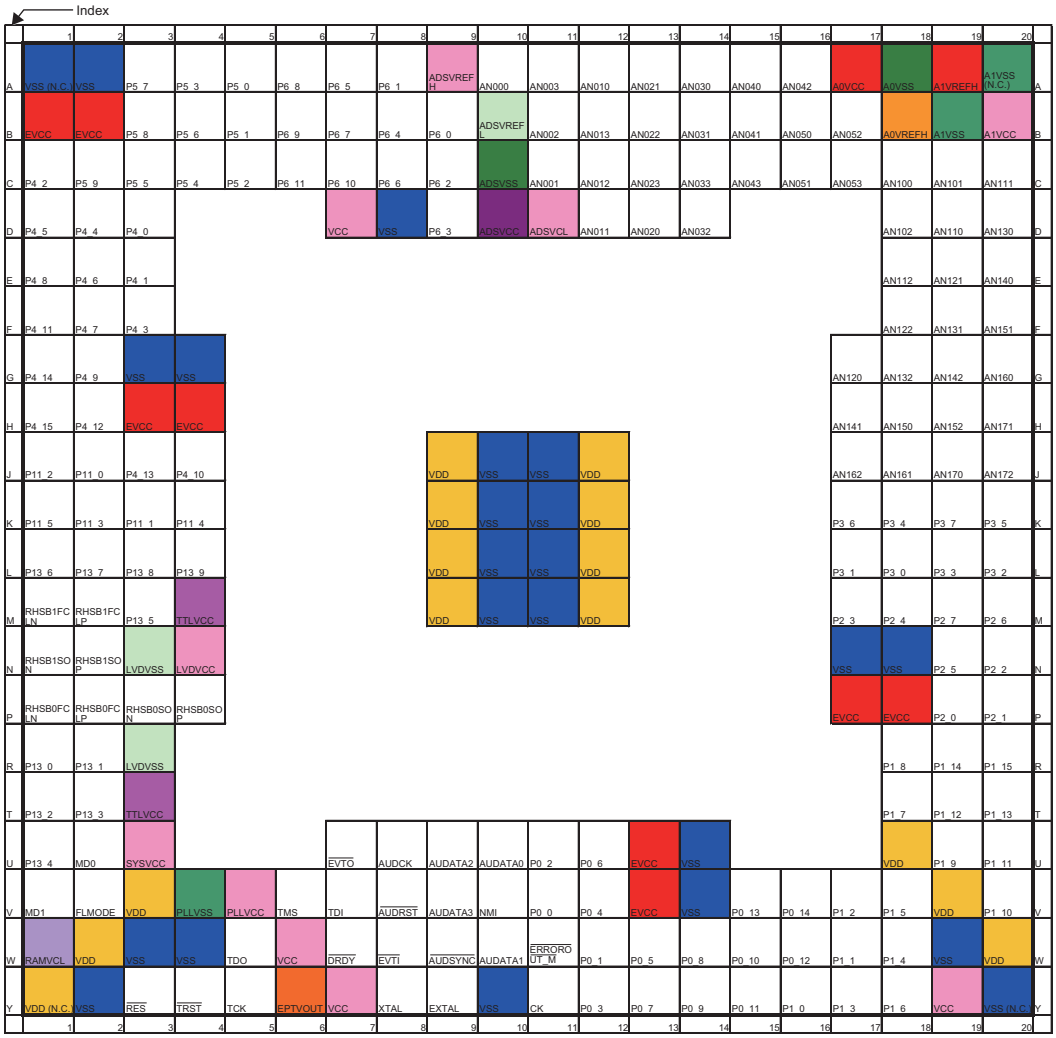


Figure 1.2 Pin Connection Diagram (E1MS-BGA252)

Table 1.1 Pin Numbers and Pin Names (E1MS-BGA304) (1/7)

Pin No.	Pin Name
A1	VSS (N.C.)
A2	VSS
A3	P6_8/TIF1A/TOD70B/TIOC70/TOE60/SCI3RxD/DSADTRG7
A4	P6_6/TOD72A/TOD62B/TIOC63/TOE52/SCI2RxD/DSADTRG6
A5	P6_3/TIF0A/TOD63A/TIOC60/TOE43/CTX3/SCI1TxD/DSADEND4
A6	P6_0/TIF13/TOD60A/TIJ0/TOE40/CRX0/DSADTRG3
A7	P8_6/TIF11/TOE22/TIJ2/TIOC72
A8	P8_4/TIF9/TOE20/TIJ0/TIOC70/SCI0TxD
A9	P8_2/TIF7/TOE12/TIOC62/SCI2TxD
A10	P10_4/TIF4/TOE03/TIOC53/CRX2/SCI1RxD
A11	P10_0/TIF0A/CRX0/SCI3RxD
A12	ADSVREFH
A13	AN000/DSAN0P
A14	AN010/DSAN2P/P14_12/TIF9
A15	AN021/DSAN4N/P14_1/TIF1A
A16	AN030/DSAN6P/P14_4/TIF1B/TIA01
A17	AN040/P14_8/TIF5/TIA04
A18	AN042/P14_10/TIF7/TIJ2
A19	A0VCC
A20	A0VSS
A21	A1VREFH
A22	A1VSS (N.C.)
B1	EVCC
B2	EVCC
B3	P6_10/TIJ4/TOD72B/TIOC72/TOE62/SCI0RxD
B4	P6_9/TIF2A/TOD71B/TIOC71/TOE61/SCI3TxD/DSADEND7
B5	P6_5/TOD71A/TOD61B/TIOC62/TOE51/CTX1/DSADEND5
B6	P6_2/TIF15/TOD62A/TIJ2/TOE42/CRX3/SCI1RxD/DSADTRG4
B7	P8_7/TIF12/TOE23/TIJ3/TIOC73/IRQ0
B8	P8_5/TIF10/TOE21/TIJ1/TIOC71
B9	P8_3/TIF8/TOE13/TIOC63/SCI0RxD
B10	P8_0/TIF5/TOE10/TIOC60/CTX2/SCI1TxD
B11	P10_1/TIF1A/TOE00/TIOC50/CTX0/SCI3TxD
B12	ADSVREFL
B13	AN003/DSAN1N
B14	AN013/DSAN3N/P14_15/TIF12/TIJ1
B15	AN022/DSAN5P/P14_2/TIF2A
B16	AN031/DSAN6N/P14_5/TIF2B/TIA02
B17	AN041/P14_9/TIF6/TIA05
B18	AN050/P15_0/TIF9/TIA06
B19	AN052/P15_2/TIF11
B20	A0VREFH
B21	A1VSS
B22	A1VCC

Table 1.1 Pin Numbers and Pin Names (E1MS-BGA304) (2/7)

Pin No.	Pin Name
C1	P5_3/TIF6/TIJ0/TOD23B/TIOC03/ESO3/CRX1/SCI2RxD
C2	P5_1/TIF1B/TOE42/TOD21B/TIOC01/CTX0/SCI1TxD
C3	P5_0/TIF0B/TOE41/TOD20B/TIOC00/CRX0/SCI1RxD
C4	P6_11/TIJ5/TOD73B/TIOC73/TOE63/SCI0TxD
C5	P6_7/TOD73A/TIJ3/TOD63B/TOE53/SCI2TxD/DSADEND6
C6	P6_4/TOD70A/TOD60B/TIOC61/TOE50/CRX1/DSADTRG5
C7	P6_1/TIF14/TOD61A/TIJ1/TOE41/CTX0/DSADEND3
C8	VSS
C9	EVCC
C10	P8_1/TIF6/TOE11/TIOC61/SCI2RxD
C11	P10_2/TIF2A/TOE01/TIOC51/CRX1
C12	ADSVSS
C13	AN001/DSAN0N
C14	AN012/DSAN3P/P14_14/TIF11
C15	AN023/DSAN5N/P14_3/TIF0B/TIA00
C16	AN033/DSAN7N/P14_7/TIF4/TIA03
C17	AN043/P14_11/TIF8/TIJ3
C18	AN051/P15_1/TIF10
C19	AN053/P15_3/TIF12
C20	AN100
C21	AN101
C22	AN111
D1	P5_6/TIF9/TOE21/TOD32B/TIOC12/LRX0/CRX2/DSADTRG0
D2	P5_4/TIF7/TIJ1/TOD30B/TIOC10/CTX1/SCI2TxD/DSADTRG1
D3	P5_2/TIF2B/TOE43/TOD22B/TIOC02/ESO2/SCI0RxD/SCI1SCK/$\overline{\text{POD}}$
D7	VCC
D8	VSS
D9	EVCC
D10	P10_3/TIF3/TOE02/TIOC52/CTX1
D11	ADSVCC
D12	ADSVCL
D13	AN002/DSAN1P
D14	AN011/DSAN2N/P14_13/TIF10/TIJ0
D15	AN020/DSAN4P/P14_0/TIF0A
D16	AN032/DSAN7P/P14_6/TIF3
D20	AN102
D21	AN110
D22	AN130
E1	P5_9/TIF12/TOE33/TIJ3/TIOC41/SCI3TxD/DSADEND2
E2	P5_8/TIF11/TOE23/TIJ2/TIOC40/SCI3RxD/DSADTRG2
E3	P5_5/TIF8/TOE20/TOD31B/TIOC11/SCI0TxD/SCI2SCK/DSADEND1
E20	AN112
E21	AN121
E22	AN140

Table 1.1 Pin Numbers and Pin Names (E1MS-BGA304) (3/7)

Pin No.	Pin Name
F1	P4_4/TOD10A/TAPTS11/TOD10B/TOE30/TIF4/APA4/CSIH2TRY/CSIH3TCSS1
F2	P4_2/TOD02A/TAPTS02/TOD02B/TIOC22/TIF2A/APA2/CSIH3TSCK
F3	P5_7/TIF10/TOE22/TOD33B/TIOC13/LTX0/CTX2/SCI3SCK/DSADEND0
F20	AN122
F21	AN131
F22	AN151/P16_1
G1	P4_7/TOD13A/TOD13B/TOE00/TIF7/APA7/CSIH2TSI/$\overline{\text{CSIH3TSSI}}$
G2	P4_5/TOD11A/TAPTS12/TOD11B/TOE31/TIF5/APA5/CRX0/CSIH3TCSS2
G3	P4_3/TOD03A/TAPTS10/TOD03B/TIOC23/TIF3/APA3/$\overline{\text{CSIH2TSSI}}$/CSIH3TCSS0
G4	P4_0/TOD00A/TAPTS00/TOD00B/TIOC20/TIF0A/APA0/$\overline{\text{ERROROUT\_C}}$/CSIH3TSI
G19	AN120
G20	AN132
G21	AN142
G22	AN160/P16_2
H1	P4_11/TOD43A/TOD43B/TOE10/TIF11/APA11/CSIH2TCSS1
H2	P4_8/TOD40A/TOD40B/TOE01/TIF8/APA8/CSIH2TSO/CSIH3TRY
H3	P4_6/TOD12A/TOD12B/TOE32/TIF6/APA6/CTX0/CSIH3TCSS3
H4	P4_1/TOD01A/TAPTS01/TOD01B/TIOC21/TIF1A/APA1/CSIH3TSO
H19	AN141
H20	AN150/P16_0
H21	AN152/P16_6
H22	AN171/P16_5
J1	P4_14/TOD20B/TOD52B/TOE13/TIF14/APA14/CRX1
J2	P4_9/TOD41A/TOD41B/TOE02/TIF9/APA9/CRX3/CSIH2TSCK
J3	VSS
J4	VSS
J9	VSS
J10	VDD
J11	VDD
J12	VDD
J13	VDD
J14	VSS
J19	AN162/P16_7
J20	AN161/P16_3
J21	AN170/P16_4
J22	AN172/P16_8
K1	P4_15/TOD21B/TOD53B/TOE40/TIF15/APA15/CTX1/$\overline{\text{ERROROUT\_C}}$
K2	P4_12/TIOC42/TOD50B/TOE11/TIF12/APA12/SCI1RxD/CSIH2TCSS2
K3	EVCC
K4	EVCC
K9	VDD
K10	VSS
K11	VSS
K12	VSS

Table 1.1 Pin Numbers and Pin Names (E1MS-BGA304) (4/7)

Pin No.	Pin Name
K13	VSS
K14	VDD
K19	VSS
K20	VSS
K21	P7_11/TOD73A/TOD73B/TIOC23
K22	P7_10/TOD72A/TOD72B/TIOC22/TOE03
L1	P11_2/TOD80A/TOD80B/APA1/TOE20
L2	P11_0
L3	P4_13/TIOC43/TOD51B/TOE12/TIF13/APA13/SCI1TxD/CSIH2TCSS3
L4	P4_10/TOD42A/TOD42B/TOE03/TIF10/APA10/CTX3/CSIH2TCSS0
L9	VDD
L10	VSS
L11	VSS
L12	VSS
L13	VSS
L14	VDD
L19	EVCC
L20	EVCC
L21	P7_9/TOD71A/TOD71B/TIOC21/TOE02
L22	P7_8/TOD70A/TOD70B/TIOC20/TOE01/CSIH3TRY
M1	P11_5/TOD83A/TOD83B/APA4/TOE23
M2	P11_3/TOD81A/TOD81B/APA2/TOE21
M3	P11_1/TIF13/TOD22B/APA0/TOE33
M4	P11_4/TOD82A/TOD82B/APA3/TOE22
M9	VDD
M10	VSS
M11	VSS
M12	VSS
M13	VSS
M14	VDD
M19	P7_4/TOD60A/TOD60B/TIOC10/CSIH3TCSS1
M20	P7_5/TOD61A/TIJ0/TOD61B/TIOC11/CSIH3TCSS2
M21	P7_7/TOD63A/TOD63B/TIOC13/TOE00/$\overline{\text{CSIH3TSSI}}$
M22	P7_6/TOD62A/TIJ1/TOD62B/TIOC12/CSIH3TCSS3
N1	P13_6/RHSB1CSD1
N2	P13_7/RHSB1SI0
N3	P13_8/RHSB1SI1
N4	P13_9/RHSB1EMRG
N9	VDD
N10	VSS
N11	VSS
N12	VSS
N13	VSS
N14	VDD

Table 1.1 Pin Numbers and Pin Names (E1MS-BGA304) (5/7)

Pin No.	Pin Name
N19	P7_2/TIOC02/CSIH3TSCK
N20	P7_0/TIOC00/CSIH3TSI
N21	P7_3/TIOC03/CSIH3TCSS0
N22	P7_1/TIOC01/CSIH3TSO
P1	RHSB1FCLN
P2	RHSB1FCLP
P3	P13_5/RHSB1CSD0
P4	TTLVCC
P9	VSS
P10	VDD
P11	VDD
P12	VDD
P13	VDD
P14	VSS
P19	P3_7/TIF6/TOD53B/TIOC01/CTX3
P20	P3_4/TIF4/TIJ2/TOD50B/TIOC42/ADTRG0/ESO3
P21	P3_6/TIA06/TOD52B/TIOC00/CRX3
P22	P3_5/TIF5/TIJ3/TOD51B/TIOC43/ADEND0
R1	RHSB1SON
R2	RHSB1SOP
R3	LVDVSS
R4	LVDVCC
R19	VSS
R20	P3_1/TIF1A/TOE61/TOD31B/TIOC33/CTX1/SCI1TxD
R21	P3_3/TIF3/TOE63/TOD33B/TIOC41/ADEND1/CTX2/ESO2/SCI2TxD
R22	P3_2/TIF2A/TOE62/TOD32B/TIOC40/ADTRG1/CRX2/SCI2RxD
T1	RHSB0FCLN
T2	RHSB0FCLP
T3	RHSB0SON
T4	RHSB0SOP
T19	EVCC
T20	P2_3/TIA03/TOE53/TOD03B/TIOC21/IRQ3
T21	P2_7/TCLKB/TOE13/TOD13B/TIOC31/CTX0/IRQ7
T22	P3_0/TIF0A/TOE60/TOD30B/TIOC32/CRX1/SCI1RxD
U1	P13_0/RHSB0CSD0/CSIH1TSI
U2	P13_1/RHSB0CSD1/CSIH1TSO
U3	LVDVSS
U20	P2_1/TIA01/TOE51/TOD01B/TIOC13/IRQ1
U21	P2_5/TIA05/TOE11/TOD11B/TIOC23/CTX3/IRQ5
U22	P2_6/TCLKA/TOE12/TOD12B/TIOC30/CRX0/IRQ6
V1	P13_2/RHSB0SI0/CSIH1TSCK
V2	P13_3/RHSB0SI1/CSIH1TCSS0
V3	TTLVCC
V20	P2_0/TIA00/TOE50/TOD00B/TIOC12/TSO17/IRQ0

Table 1.1 Pin Numbers and Pin Names (E1MS-BGA304) (6/7)

Pin No.	Pin Name
V21	P2_4/TIA04/TOE10/TOD10B/TIOC22/CRX3/IRQ4
V22	P2_2/TIA02/TOE52/TOD02B/TIOC20/IRQ2
W1	P13_4/RHSB0EMRG/CSIH1TCSS1/$\overline{\text{ERROROUT\_C}}$
W2	MD0
W3	SYSVCC
W7	$\overline{\text{EVTO}}$
W8	AUDCK
W9	AUDATA2
W10	AUDATA0
W11	P0_2/$\overline{\text{FR0ENA}}$/CSIH0TSCK/SCI0SCK
W12	P0_6/CSIH1TSI/IRQ3
W13	EVCC
W14	VSS
W15	P0_12/TIA04/TOD01B/CSIH0TCSS4/CSIH1TCSS3/CSIH2TSO
W16	P1_6/TOD32A/TOD32B/TOE42/APA6/TSO05
W20	VDD
W21	P1_15/TIOC53/TOD63B/TOE33/APA15/TSO16
W22	P1_14/TIOC52/TOD62B/TOE32/APA14/TSO15
Y1	MD1
Y2	FLMODE
Y3	VDD
Y4	PLLVSS
Y5	PLLVCC
Y6	TMS/$\overline{\text{EVTO}}$
Y7	TDI/LPDI/FLSCI3RX
Y8	$\overline{\text{AUDRST}}$
Y9	AUDATA3
Y10	NMI
Y11	P0_0/FR0RXDA/CSIH0TSI/SCI0RxD
Y12	P0_4/FR0TXDB/CSIH0TCSS1/CTX2/IRQ1
Y13	EVCC
Y14	VSS
Y15	P0_14/TIA06/TOD03B/CSIH1TRY/CSIH2TCSS0
Y16	P1_3/TOD23A/TOD23B/TOE03/APA3/TSO02/$\overline{\text{CSIH2TSSI}}$
Y17	P1_8/TOD50A/TOD50B/TOE20/APA8/TSO07/LRX0
Y18	P1_10/TOD52A/TOD52B/TOE22/APA10/TSO11
Y19	P1_12/TIOC50/TOD60B/TOE30/APA12/TSO13
Y20	VSS
Y21	VDD
Y22	VCC
AA1	RAMVCL
AA2	VDD
AA3	VSS
AA4	VSS

Table 1.1 Pin Numbers and Pin Names (E1MS-BGA304) (7/7)

Pin No.	Pin Name
AA5	TDO/LPDO/FLSCI3TX
AA6	VCC
AA7	$\overline{\text{DRDY}}$/LPDCLKO
AA8	$\overline{\text{EVTI}}$
AA9	$\overline{\text{AUDSYNC}}$
AA10	AUDATA1
AA11	$\overline{\text{ERROROUT\_M}}$
AA12	P0_1/FR0TXDA/CSIH0TSO/SCI0TxD
AA13	P0_5/$\overline{\text{FR0ENB}}$/CSIH0TCSS2/IRQ2
AA14	P0_8/TIA00/CSIH1TSCK/IRQ5
AA15	P0_10/TIA02/CSIH0TRY/CSIH1TCSS1/IRQ7
AA16	P1_0/TOD20A/TOD20B/TOE00/APA0/SCI0RxD/CSIH2TCSS1
AA17	P1_2/TOD22A/TOD22B/TOE02/APA2/TSO01/CSIH2TCSS3
AA18	P1_4/TOD30A/TOD30B/TOE40/APA4/TSO03/CSIH2TRY
AA19	P1_7/TOD33A/TOD33B/TOE43/APA7/TSO06
AA20	P1_11/TOD53A/TOD53B/TOE23/APA11/TSO12
AA21	VSS
AA22	VDD
AB1	VDD (N.C.)
AB2	VSS
AB3	$\overline{\text{RES}}$
AB4	$\overline{\text{TRST}}$/$\overline{\text{LPDRST}}$
AB5	TCK/LPDCLKI/FLSCI3SCK
AB6	EPTVOUT
AB7	VCC
AB8	XTAL
AB9	EXTAL
AB10	VSS
AB11	CK
AB12	P0_3/FR0RXDB/CSIH0TCSS0/CRX2/IRQ0
AB13	P0_7/CSIH1TSO/IRQ4
AB14	P0_9/TIA01/$\overline{\text{CSIH0TSSI}}$/CSIH1TCSS0/IRQ6
AB15	P0_11/TIA03/TOD00B/CSIH0TCSS3/CSIH1TCSS2/CSIH2TSI
AB16	P0_13/TIA05/TOD02B/CSIH0TCSS5/$\overline{\text{CSIH1TSSI}}$/CSIH2TSCK
AB17	P1_1/TOD21A/TOD21B/TOE01/APA1/TSO00/SCI0TxD/CSIH2TCSS2
AB18	P1_5/TOD31A/TOD31B/TOE41/APA5/TSO04
AB19	P1_9/TOD51A/TOD51B/TOE21/APA9/TSO10/LTX0
AB20	P1_13/TIOC51/TOD61B/TOE31/APA13/TSO14
AB21	EVCC
AB22	VSS (N.C.)

Table 1.2 Pin Numbers and Pin Names (E1MS-BGA252) (1/6)

Pin No.	Pin Name
A1	VSS (N.C.)
A2	VSS
A3	P5_7/TIF10/TOE22/TOD33B/TIOC13/LTX0/CTX2/SCI3SCK/DSADEND0
A4	P5_3/TIF6/TIJ0/TOD23B/TIOC03/ESO3/CRX1/SCI2RxD
A5	P5_0/TIF0B/TOE41/TOD20B/TIOC00/CRX0/SCI1RxD
A6	P6_8/TIF1A/TOD70B/TIOC70/TOE60/SCI3RxD/DSADTRG7
A7	P6_5/TOD71A/TOD61B/TIOC62/TOE51/CTX1/DSADEND5
A8	P6_1/TIF14/TOD61A/TIJ1/TOE41/CTX0/DSADEND3
A9	ADSVREFH
A10	AN000/DSAN0P
A11	AN003/DSAN1N
A12	AN010/DSAN2P/P14_12/TIF9
A13	AN021/DSAN4N/P14_1/TIF1A
A14	AN030/DSAN6P/P14_4/TIF1B/TIA01
A15	AN040/P14_8/TIF5/TIA04
A16	AN042/P14_10/TIF7/TIJ2
A17	A0VCC
A18	A0VSS
A19	A1VREFH
A20	A1VSS (N.C.)
B1	EVCC
B2	EVCC
B3	P5_8/TIF11/TOE23/TIJ2/TIOC40/SCI3RxD/DSADTRG2
B4	P5_6/TIF9/TOE21/TOD32B/TIOC12/LRX0/CRX2/DSADTRG0
B5	P5_1/TIF1B/TOE42/TOD21B/TIOC01/CTX0/SCI1TxD
B6	P6_9/TIF2A/TOD71B/TIOC71/TOE61/SCI3TxD/DSADEND7
B7	P6_7/TOD73A/TIJ3/TOD63B/TOE53/SCI2TxD/DSADEND6
B8	P6_4/TOD70A/TOD60B/TIOC61/TOE50/CRX1/DSADTRG5
B9	P6_0/TIF13/TOD60A/TIJ0/TOE40/CRX0/DSADTRG3
B10	ADSVREFL
B11	AN002/DSAN1P
B12	AN013/DSAN3N/P14_15/TIF12/TIJ1
B13	AN022/DSAN5P/P14_2/TIF2A
B14	AN031/DSAN6N/P14_5/TIF2B/TIA02
B15	AN041/P14_9/TIF6/TIA05
B16	AN050/P15_0/TIF9/TIA06
B17	AN052/P15_2/TIF11
B18	A0VREFH
B19	A1VSS
B20	A1VCC
C1	P4_2/TOD02A/TAPTS02/TOD02B/TIOC22/TIF2A/APA2/CSIH3TSCK
C2	P5_9/TIF12/TOE33/TIJ3/TIOC41/SCI3TxD/DSADEND2
C3	P5_5/TIF8/TOE20/TOD31B/TIOC11/SCI0TxD/SCI2SCK/DSADEND1
C4	P5_4/TIF7/TIJ1/TOD30B/TIOC10/CTX1/SCI2TxD/DSADTRG1

Table 1.2 Pin Numbers and Pin Names (E1MS-BGA252) (2/6)

Pin No.	Pin Name
C5	P5_2/TIF2B/TOE43/TOD22B/TIOC02/ESO2/SCI0RxD/SCI1SCK/$\overline{\text{POD}}$
C6	P6_11/TIJ5/TOD73B/TIOC73/TOE63/SCI0TxD
C7	P6_10/TIJ4/TOD72B/TIOC72/TOE62/SCI0RxD
C8	P6_6/TOD72A/TOD62B/TIOC63/TOE52/SCI2RxD/DSADTRG6
C9	P6_2/TIF15/TOD62A/TIJ2/TOE42/CRX3/SCI1RxD/DSADTRG4
C10	ADSVSS
C11	AN001/DSAN0N
C12	AN012/DSAN3P/P14_14/TIF11
C13	AN023/DSAN5N/P14_3/TIF0B/TIA00
C14	AN033/DSAN7N/P14_7/TIF4/TIA03
C15	AN043/P14_11/TIF8/TIJ3
C16	AN051/P15_1/TIF10
C17	AN053/P15_3/TIF12
C18	AN100
C19	AN101
C20	AN111
D1	P4_5/TOD11A/TAPTS12/TOD11B/TOE31/TIF5/APA5/CRX0/CSIH3TCSS2
D2	P4_4/TOD10A/TAPTS11/TOD10B/TOE30/TIF4/APA4/CSIH2TRY/CSIH3TCSS1
D3	P4_0/TOD00A/TAPTS00/TOD00B/TIOC20/TIF0A/APA0/$\overline{\text{ERROROUT\_C}}$/CSIH3TSI
D7	VCC
D8	VSS
D9	P6_3/TIF0A/TOD63A/TIOC60/TOE43/CTX3/SCI1TxD/DSADEND4
D10	ADSVCC
D11	ADSVCL
D12	AN011/DSAN2N/P14_13/TIF10/TIJ0
D13	AN020/DSAN4P/P14_0/TIF0A
D14	AN032/DSAN7P/P14_6/TIF3
D18	AN102
D19	AN110
D20	AN130
E1	P4_8/TOD40A/TOD40B/TOE01/TIF8/APA8/CSIH2TSO/CSIH3TRY
E2	P4_6/TOD12A/TOD12B/TOE32/TIF6/APA6/CTX0/CSIH3TCSS3
E3	P4_1/TOD01A/TAPTS01/TOD01B/TIOC21/TIF1A/APA1/CSIH3TSO
E18	AN112
E19	AN121
E20	AN140
F1	P4_11/TOD43A/TOD43B/TOE10/TIF11/APA11/CSIH2TCSS1
F2	P4_7/TOD13A/TOD13B/TOE00/TIF7/APA7/CSIH2TSI/$\overline{\text{CSIH3TSSI}}$
F3	P4_3/TOD03A/TAPTS10/TOD03B/TIOC23/TIF3/APA3/$\overline{\text{CSIH2TSSI}}$/CSIH3TCSS0
F18	AN122
F19	AN131
F20	AN151/P16_1
G1	P4_14/TOD20B/TOD52B/TOE13/TIF14/APA14/CRX1
G2	P4_9/TOD41A/TOD41B/TOE02/TIF9/APA9/CRX3/CSIH2TSCK

Table 1.2 Pin Numbers and Pin Names (E1MS-BGA252) (3/6)

Pin No.	Pin Name
G3	VSS
G4	VSS
G17	AN120
G18	AN132
G19	AN142
G20	AN160/P16_2
H1	P4_15/TOD21B/TOD53B/TOE40/TIF15/APA15/CTX1/$\overline{\text{ERROROUT\_C}}$
H2	P4_12/TIOC42/TOD50B/TOE11/TIF12/APA12/SCI1RxD/CSIH2TCSS2
H3	EVCC
H4	EVCC
H17	AN141
H18	AN150/P16_0
H19	AN152/P16_6
H20	AN171/P16_5
J1	P11_2/TOD80A/TOD80B/APA1/TOE20
J2	P11_0
J3	P4_13/TIOC43/TOD51B/TOE12/TIF13/APA13/SCI1TxD/CSIH2TCSS3
J4	P4_10/TOD42A/TOD42B/TOE03/TIF10/APA10/CTX3/CSIH2TCSS0
J9	VDD
J10	VSS
J11	VSS
J12	VDD
J17	AN162/P16_7
J18	AN161/P16_3
J19	AN170/P16_4
J20	AN172/P16_8
K1	P11_5/TOD83A/TOD83B/APA4/TOE23
K2	P11_3/TOD81A/TOD81B/APA2/TOE21
K3	P11_1/TIF13/TOD22B/APA0/TOE33
K4	P11_4/TOD82A/TOD82B/APA3/TOE22
K9	VDD
K10	VSS
K11	VSS
K12	VDD
K17	P3_6/TIA06/TOD52B/TIOC00/CRX3
K18	P3_4/TIF4/TIJ2/TOD50B/TIOC42/ADTRG0/ESO3
K19	P3_7/TIF6/TOD53B/TIOC01/CTX3
K20	P3_5/TIF5/TIJ3/TOD51B/TIOC43/ADEND0
L1	P13_6/RHSB1CSD1
L2	P13_7/RHSB1SI0
L3	P13_8/RHSB1SI1
L4	P13_9/RHSB1EMRG
L9	VDD
L10	VSS

Table 1.2 Pin Numbers and Pin Names (E1MS-BGA252) (4/6)

Pin No.	Pin Name
L11	VSS
L12	VDD
L17	P3_1/TIF1A/TOE61/TOD31B/TIOC33/CTX1/SCI1TxD
L18	P3_0/TIF0A/TOE60/TOD30B/TIOC32/CRX1/SCI1RxD
L19	P3_3/TIF3/TOE63/TOD33B/TIOC41/ADEND1/CTX2/ESO2/SCI2TxD
L20	P3_2/TIF2A/TOE62/TOD32B/TIOC40/ADTRG1/CRX2/SCI2RxD
M1	RHSB1FCLN
M2	RHSB1FCLP
M3	P13_5/RHSB1CSD0
M4	TTLVCC
M9	VDD
M10	VSS
M11	VSS
M12	VDD
M17	P2_3/TIA03/TOE53/TOD03B/TIOC21/IRQ3
M18	P2_4/TIA04/TOE10/TOD10B/TIOC22/CRX3/IRQ4
M19	P2_7/TCLKB/TOE13/TOD13B/TIOC31/CTX0/IRQ7
M20	P2_6/TCLKA/TOE12/TOD12B/TIOC30/CRX0/IRQ6
N1	RHSB1SON
N2	RHSB1SOP
N3	LVDVSS
N4	LVDVCC
N17	VSS
N18	VSS
N19	P2_5/TIA05/TOE11/TOD11B/TIOC23/CTX3/IRQ5
N20	P2_2/TIA02/TOE52/TOD02B/TIOC20/IRQ2
P1	RHSB0FCLN
P2	RHSB0FCLP
P3	RHSB0SON
P4	RHSB0SOP
P17	EVCC
P18	EVCC
P19	P2_0/TIA00/TOE50/TOD00B/TIOC12/TSO17/IRQ0
P20	P2_1/TIA01/TOE51/TOD01B/TIOC13/IRQ1
R1	P13_0/RHSB0CSD0/CSIH1TSI
R2	P13_1/RHSB0CSD1/CSIH1TSO
R3	LVDVSS
R18	P1_8/TOD50A/TOD50B/TOE20/APA8/TSO07/LRX0
R19	P1_14/TIOC52/TOD62B/TOE32/APA14/TSO15
R20	P1_15/TIOC53/TOD63B/TOE33/APA15/TSO16
T1	P13_2/RHSB0SI0/CSIH1TSCK
T2	P13_3/RHSB0SI1/CSIH1TCSS0
T3	TTLVCC
T18	P1_7/TOD33A/TOD33B/TOE43/APA7/TSO06

Table 1.2 Pin Numbers and Pin Names (E1MS-BGA252) (5/6)

Pin No.	Pin Name
T19	P1_12/TIOC50/TOD60B/TOE30/APA12/TSO13
T20	P1_13/TIOC51/TOD61B/TOE31/APA13/TSO14
U1	P13_4/RHSB0EMRG/CSIH1TCSS1/$\overline{\text{ERROROUT\_C}}$
U2	MD0
U3	SYSVCC
U7	$\overline{\text{EVTO}}$
U8	AUDCK
U9	AUDATA2
U10	AUDATA0
U11	P0_2/$\overline{\text{FR0ENA}}$/CSIH0TSCK/SCI0SCK
U12	P0_6/CSIH1TSI/IRQ3
U13	EVCC
U14	VSS
U18	VDD
U19	P1_9/TOD51A/TOD51B/TOE21/APA9/TSO10/LTX0
U20	P1_11/TOD53A/TOD53B/TOE23/APA11/TSO12
V1	MD1
V2	FLMODE
V3	VDD
V4	PLLVSS
V5	PLLVCC
V6	TMS/$\overline{\text{EVTO}}$
V7	TDI/LPDI/FLSCI3RX
V8	$\overline{\text{AUDRST}}$
V9	AUDATA3
V10	NMI
V11	P0_0/FR0RXDA/CSIH0TSI/SCI0RxD
V12	P0_4/FR0TXDB/CSIH0TCSS1/CTX2/IRQ1
V13	EVCC
V14	VSS
V15	P0_13/TIA05/TOD02B/CSIH0TCSS5/$\overline{\text{CSIH1TSSI}}$/CSIH2TSCK
V16	P0_14/TIA06/TOD03B/CSIH1TRY/CSIH2TCSS0
V17	P1_2/TOD22A/TOD22B/TOE02/APA2/TSO01/CSIH2TCSS3
V18	P1_5/TOD31A/TOD31B/TOE41/APA5/TSO04
V19	VDD
V20	P1_10/TOD52A/TOD52B/TOE22/APA10/TSO11
W1	RAMVCL
W2	VDD
W3	VSS
W4	VSS
W5	TDO/LPDO/FLSCI3TX
W6	VCC
W7	$\overline{\text{DRDY}}$/LPDCLKO
W8	$\overline{\text{EVTI}}$

Table 1.2 Pin Numbers and Pin Names (E1MS-BGA252) (6/6)

Pin No.	Pin Name
W9	$\overline{\text{AUDSYNC}}$
W10	AUDATA1
W11	$\overline{\text{ERROROUT\_M}}$
W12	P0_1/FR0TXDA/CSIH0TSO/SCI0TxD
W13	P0_5/$\overline{\text{FR0ENB}}$/CSIH0TCSS2/IRQ2
W14	P0_8/TIA00/CSIH1TSCK/IRQ5
W15	P0_10/TIA02/CSIH0TRY/CSIH1TCSS1/IRQ7
W16	P0_12/TIA04/TOD01B/CSIH0TCSS4/CSIH1TCSS3/CSIH2TSO
W17	P1_1/TOD21A/TOD21B/TOE01/APA1/TSO00/SCI0TxD/CSIH2TCSS2
W18	P1_4/TOD30A/TOD30B/TOE40/APA4/TSO03/CSIH2TRY
W19	VSS
W20	VDD
Y1	VDD (N.C.)
Y2	VSS
Y3	$\overline{\text{RES}}$
Y4	$\overline{\text{TRST}}$/$\overline{\text{LPDRST}}$
Y5	TCK/LPDCLKI/FLSCI3SCK
Y6	EPTVOUT
Y7	VCC
Y8	XTAL
Y9	EXTAL
Y10	VSS
Y11	CK
Y12	P0_3/FR0RXDB/CSIH0TCSS0/CRX2/IRQ0
Y13	P0_7/CSIH1TSO/IRQ4
Y14	P0_9/TIA01/$\overline{\text{CSIH0TSSI}}$/CSIH1TCSS0/IRQ6
Y15	P0_11/TIA03/TOD00B/CSIH0TCSS3/CSIH1TCSS2/CSIH2TSI
Y16	P1_0/TOD20A/TOD20B/TOE00/APA0/SCI0RxD/CSIH2TCSS1
Y17	P1_3/TOD23A/TOD23B/TOE03/APA3/TSO02/$\overline{\text{CSIH2TSSI}}$
Y18	P1_6/TOD32A/TOD32B/TOE42/APA6/TSO05
Y19	VCC
Y20	VSS (N.C.)

1.6 Functional Block Configuration

1.6.1 Internal Block Diagram

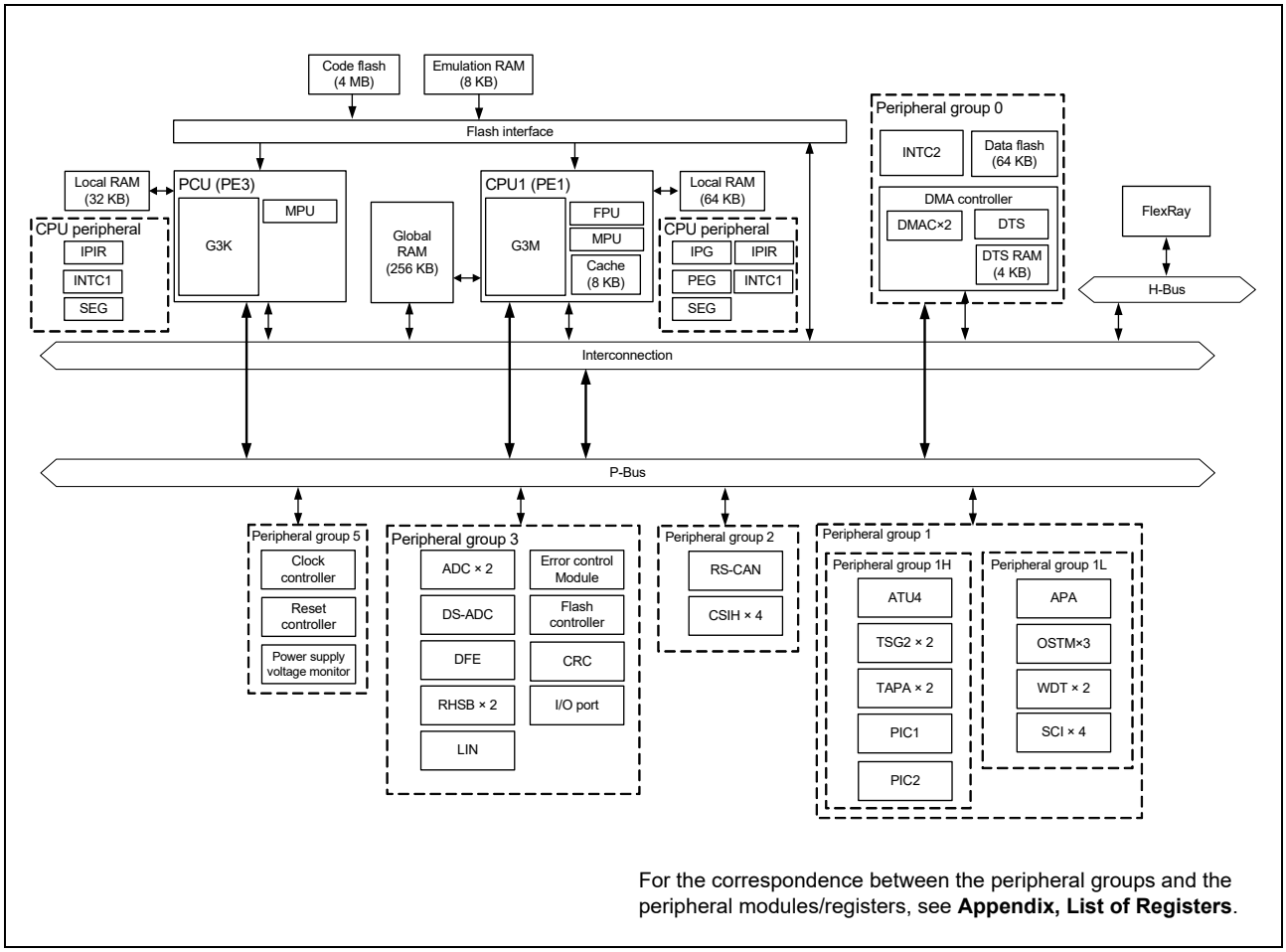


Figure 1.3 Internal Block Diagram

The CPU1 and the PCU incorporate the CPU peripherals. Only the CPU and the PCU can access the CPU peripherals. Same addresses are allocated to peripherals specific to the CPU1 and the PCU. When an access is made from the CPU1, a CPU1-specific peripheral is accessed. When an access is made from the PCU, a PCU-specific peripheral is accessed.

Section 2 Pins

2.1 Port Functions

2.1.1 Features

Port Group

This product has the following numbers of port groups:

Table 2.1 Port Groups of This Product

Product		Number of Group	Name of Group
E1M-S	BGA304	16	P0 - P8, P10, P11, P13 - P17
	BGA252	13	P0 - P6, P11, P13 - P17

Port Group Index n

Each port group is identified by its own index “n” (n = 0 to 17) throughout this section; e.g. PMCn for the port mode control register of the Pn pin.

Register Address

All port addresses are given as an offset from the individual base addresses, <PORT_Base2> and <PORT_Base1>.

Table 2.2 shows the base addresses, <PORT_Base2> and <PORT_Base1>.

Table 2.2 Port Base Address

Base Address	Address
<PORT_Base1>	FF61 0000_H
<PORT_Base2>	FFC1 0000_H

2.1.2 Overview

This product has various pins for input/output (I/O) functions, known as ports. The ports are organized in port groups.

This product also has several control registers to allocate the functions other than general I/O purpose to the corresponding pins.

For definitions of pin, port, and port group, see **Section 2.1.2.1, Terms**.

2.1.2.1 Terms

The terms described in this section are defined as follows.

- Port group:
  One port has up to 16 pins, and the number of pins differs according to the port. The multiple pins of a port share the same control registers.
- Port mode / Port:
  In port mode, a pin functions as a general-purpose I/O pin. The general-purpose I/O function is called a “port pin” and is indicated by the name Pn_m. For example, P0_7 indicates pin 7 of port 0.
- Alternative mode:
  In alternative mode, a pin functions as an I/O pin of a peripheral function. Multiple peripheral functions may be multiplexed on a pin, and control registers select the peripheral function to be used.

2.1.2.2 Overview of Pin Functions

The pin can operate in the following three different modes:

- Port mode (PMCn.PMCn_m = 0)
  The pin operates as a general purpose I/O port in port mode.
  PMn.PMn_m selects input or output.
- S/W I/O control alternative mode (PMCn.PMCn_m = 1, PIPCn.PIPCn_m = 0)
  The pin is operated by an alternative function in S/W I/O control alternative mode.
  The selection between input and output is made by S/W via the PMn.PMn_m control bits.
- Direct I/O control alternative mode (PMCn.PMCn_m = 1, PIPCn.PIPCn_m = 1)
  The pin is operated by an alternative function in direct I/O control alternative mode.
  In contrast to S/W I/O control alternative mode, input/out is directly controlled by the alternative function in this mode.

Table 2.3 shows the outline of the register settings.

Table 2.3 Pin Function Configuration (Outline)

Mode	Bit			I/O
	PMCn_m	PMn_m	PIPCn_m	
Port mode	0	0	0/1	Output mode
		1	0/1	Input mode*1
S/W I/O control alternative mode	1	0	0	Output mode
		1	0	Input mode*2
Direct I/O control alternative mode		0/1	1	Controlled by the alternative function*2

Note 1. The input buffer should be enabled (PIBCn_m = 1).
Note 2. When used as an input pin in alternative mode, be sure to set PIBCn_m = 0.

When a pin is operated in alternative mode (PMCn.PMCn_m = 1), one of different alternative functions (up to 7) can be selected by the PFCn, PFCEn, and PFCAEn registers.

- S/W I/O control alternative mode (PIPCn.PIPCn_m = 0):
  - Outputs (PMn_m = 0): ALT-OUT1 to ALT-OUT7
  - Inputs (PMn_m = 1): ALT-IN1 to ALT-IN7
- Direct I/O control alternative mode (PIPCn.PIPCn_m = 1):
  - Input/Output of ALT-OUT1 to ALT-OUT7 and ALT-IN1 to ALT-IN7 is directly controlled by the alternative function.

Table 2.4 Outline of Alternative Mode Selection (PMCn.PMCn_m = 1)

Function	Register				I/O
	PFCAE	PFCE	PFC	PM*1	
Alternative output mode 1 (ALT-OUT1)	0	0	0	0	O
Alternative input mode 1 (ALT-IN1)	0	0	0	1	I
Alternative output mode 2 (ALT-OUT2)	0	0	1	0	O
Alternative input mode 2 (ALT-IN2)	0	0	1	1	I
Alternative output mode 3 (ALT-OUT3)	0	1	0	0	O
Alternative input mode 3 (ALT-IN3)	0	1	0	1	I
Alternative output mode 4 (ALT-OUT4)	0	1	1	0	O
Alternative input mode 4 (ALT-IN4)	0	1	1	1	I
Alternative output mode 5 (ALT-OUT5)	1	0	0	0	O
Alternative input mode 5 (ALT-IN5)	1	0	0	1	I
Alternative output mode 6 (ALT-OUT6)	1	0	1	0	O
Alternative input mode 6 (ALT-IN6)	1	0	1	1	I
Alternative output mode 7 (ALT-OUT7)	1	1	0	0	O
Alternative input mode 7 (ALT-IN7)	1	1	0	1	I

Note 1. When PIPCn.PIPCn_m = 1, the I/O direction is directly controlled by the peripheral (alternative) function and PM is ignored.

When a pin is operated in alternative mode (PMCn.PMCn_m = 1), one of several alternative functions can be selected by the PFCn, PFCEn, and PFCAEn registers.

2.1.2.3 Pin Data Input/Output

The registers used for data input and output are described below.

The source of the data to be read via the PPRn register depends on pin mode.

Output data

In port mode (PMCn.PMCn_m = 0), the value of Pn.Pn_m is output from the Pn_m pin.

Input data

A read operation of the PPRn register returns either the value of the Pn_m pin, the associated bit of the port register Pn.Pn_m, or the data output by an alternative function.

The source of the data read via PPRn depends on pin mode and setting of several control bits.

Table 2.5 summarizes the differences of PPRn read modes.

Table 2.5 PPRn_m Read Values

PMC n_m	PM n_m	PIBC n_m	PIPC n_m	Mode	PPRn_m Read Value
0	1	0	X	Port input, input buffer disabled	Pn.Pn_m bit
		1		Port input, input buffer enabled	Pn_m pin
	0	X		Port push-pull output	Pn.Pn_m bit*1
1	1	X	0	S/W I/O control alternative input	Pn_m pin
	0			S/W I/O control alternative output	Alternative function internal output signal*1
	X		1	Direct I/O control alternative mode	I/O port in alternative mode: • Input: Pn_m pin • Output: alternative function internal output signal*1

Note 1. When PBDCn_m = 1, Pn_m pin level is read via PPRn_m bit.

The control registers in **Table 2.5** have the following effects:

- PMCn.PMCn_m
  This bit selects either port mode (PMCn_m = 0) or alternative mode (PMCn_m = 1).
- PMn.PMn_m
  This bit selects input (PMn_m = 1) or output (PMn_m = 0) in port mode (PMCn_m = 0) and S/W I/O control alternative mode (PMCn_m = 1, PIPCn_m = 0).
- PIBCn.PIBCn_m
  This bit disables (PIBCn_m = 0) or enables (PIBCn_m = 1) the input buffer in input port mode (PMCn_m = 0 and PMn_m = 1). When the input buffer is disabled, PPRn_m reads the Pn.Pn_m bit, otherwise the Pn_m pin level is returned.
- PIPCn.PIPCn_m
  This bit selects either the S/W or direct I/O control alternative mode.
- PBDCn.PBDCn_m
  Setting this bit to 1 in output mode enables bidirectional mode for the pin. In bidirectional mode, the level on pin Pn_m can be read from PPPn.PPRn_m.

Write to the Pn Register

The data to be output via port Pn_m in port mode (PMCn.PMCn_m = 0) is held in the port register Pn.

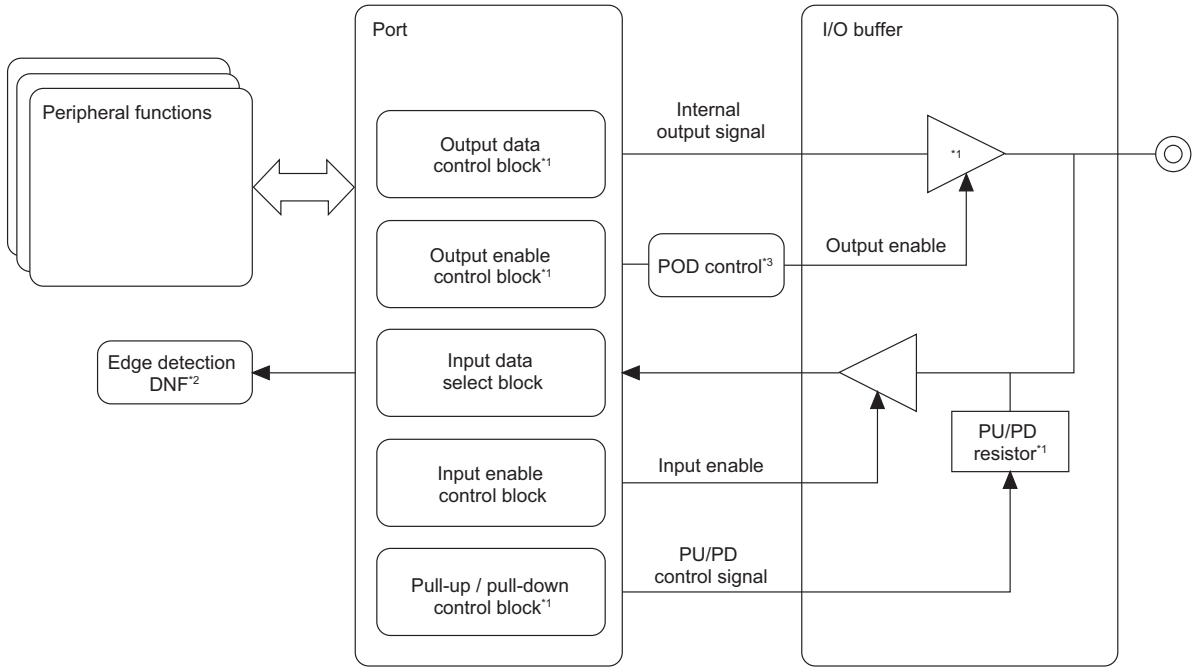
The Pn data can be rewritten in the following two different ways:

- Direct write to the Pn register
  New data can be directly written to the Pn register.
- Indirect operation to the Pn bit (set/reset/not)
  The indirect Pn operation is possible using the following two registers:
  - Port set/reset register: PSRn
    When the bit PSRn.PSRn_(m + 16) = 1, the value of bit PSRn.PSRn_m determines the value of Pn.Pn_m.
    Thus Pn_m can be set/reset without a direct write to Pn.
  - Port NOT register: PNOTn
    Setting PNOTn.PNOTn_m = 1 inverts the bit Pn.Pn_m without a direct write to Pn.

The indirect Pn set/reset/not operation provides access to single bits (not limited to one bit) of the Pn register while leaving all other Pn bits untouched.

2.1.3 Port Type

Figure 2.1 shows the overall configuration of the pins. For the details of port blocks, see **Figure 2.2**.



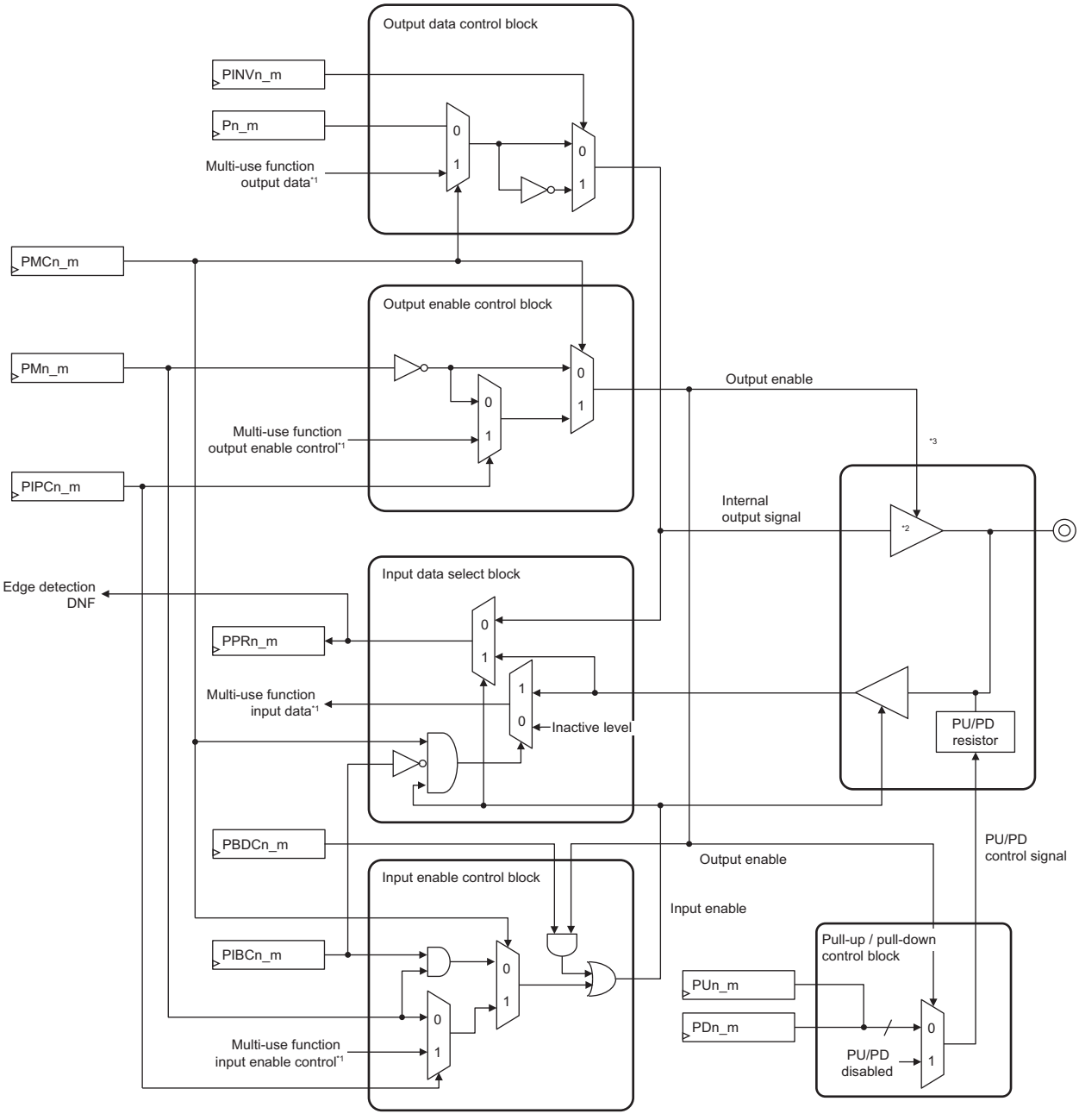
Note 1. Pins of ports P14, P15, and P16 have multiplexed analog input functions and, because these pins are only for input, they do not incorporate output-related functions. And these port groups do not incorporate pull-up and pull-down resistor.

Note 2. For details of the edge-detection DNF function and the relevant pins, see **Section 2.3.3, Edge Detection DNF**.

Note 3. For details of the POD control function and the relevant pins, see **Section 2.4, POD Control**.

Figure 2.1 Block Diagram of Pin Configuration

Figure 2.2 shows the logical circuitry of the port control functions. The diagram is only a logical reference and does not show the real circuitry.



Note 1. Multi-use function signal selected by PFCn_m, PFCEn_m, and PFCAEn_m
Note 2. Output drive strength can be set by PDSCn_m and PUCCn_m.
Note 3. The target pin of POD control forcibly disables the enabling of output when POD control is enabled.

Figure 2.2 Port Control Logic Diagram

2.1.4 Port Group Configuration Register

This section starts with an overview of all configuration registers and then presents all registers in detail. The configuration registers are classified as follows:

- **2.1.4.2, Configuration of Pin Function**
- **2.1.4.3, Pin Data Input/Output**
- **2.1.4.4, Configuration of Electrical Characteristics**
- **2.1.4.5, Port Register Protection**
- **2.1.4.6, Pin-Unit Register**

2.1.4.1 Outline

The following registers are used for the configuration of the individual pins of the port groups:

Table 2.6 Registers for Port Group Configuration

Register Name	Symbol	Register Configuration Unit	Address
Port register	Pn	Port group	<PORT_Base1> + 0000_H + n × 40_H
Port set/reset register	PSRn	Port group	<PORT_Base1> + 0004_H + n × 40_H
Port NOT register	PNOTn	Port group	<PORT_Base1> + 0008_H + n × 40_H
Port pin read register	PPRn	Port group	<PORT_Base1> + $000C_H$ + n × 40_H
Port mode register	PMn	Port group	<PORT_Base1> + 0010_H + n × 40_H
Port mode control register	PMCn	Port group	<PORT_Base1> + 0014_H + n × 40_H
Port function control register	PFCn	Port group	<PORT_Base1> + 0018_H + n × 40_H
Port function control expansion register	PFCEn	Port group	<PORT_Base1> + $001C_H$ + n × 40_H
Port mode set/reset register	PMSRn	Port group	<PORT_Base1> + 0020_H + n × 40_H
Port mode control set/reset register	PMCSRn	Port group	<PORT_Base1> + 0024_H + n × 40_H
Port function control additional expansion register	PFCAEn	Port group	<PORT_Base1> + 0028_H + n × 40_H
Port output level inversion register	PINVn	Port group	<PORT_Base1> + 0030_H + n x 40_H
Port input buffer control register	PIBCn	Port group	<PORT_Base2> + 4000_H + n × 40_H
Port bidirectional control register	PBDCn	Port group	<PORT_Base2> + 4004_H + n × 40_H
Port IP control register	PIPCn	Port group	<PORT_Base2> + 4008_H + n × 40_H
Pull-up option register	PUn	Port group	<PORT_Base2> + $400C_H$ + n × 40_H
Pull-down option register	PDn	Port group	<PORT_Base2> + 4010_H + n × 40_H
Port drive strength control register	PDSCn	Port group	<PORT_Base2> + 4018_H + n × 40_H
Port universal control register	PUCCn	Port group	<PORT_Base2> + 4028_H+ n × 40_H
Port register protection command register	PPCMDn	Port group	<PORT_Base2> + $402C_H$ + n × 40_H
Port protection status register	PPROTSn	Port group	<PORT_Base2> + 4034_H + n × 40_H
Port control register	PCRn_m	Pin	<PORT_Base1> + 2000_H + n × 40_H + m × 4_H

Note: n: Port group number
m: Bit number in a port group

Base address

The PORTn base addresses, <PORT_Base2> and <PORT_Base1> are defined in Register Addresses, **Section 2.1.1, Features**.

Register value after reset

The values after reset release depend on the port, and are not described in the following register descriptions, but are given in **Section 2.2.1.1, List of the E1M-S Port Registers**.

2.1.4.2 Configuration of Pin Function

(1) PMCn — Port Mode Control Register

This register specifies whether the individual pins of port group n are in port mode or in alternative mode.

Access: This register is readable/writable in 16-bit units.

Value after reset: See **Section 2.2.1.1, List of the E1M-S Port Registers**.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PMCn_15	PMCn_14	PMCn_13	PMCn_12	PMCn_11	PMCn_10	PMCn_9	PMCn_8	PMCn_7	PMCn_6	PMCn_5	PMCn_4	PMCn_3	PMCn_2	PMCn_1	PMCn_0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 2.7 PMCn Register Contents

Bit Position	Bit Name	Function
15 to 0	PMCn_[15:0]	Specify the operation mode of the corresponding pin: 0: Port mode 1: Alternative mode

(2) PMCSRn — Port Mode Control Set/Reset Register

This register provides an alternative method to write data to a bit in the PMCn register.

The 16 higher-order bits of PMCSRn specify whether the data in the 16 lower-order bits of PMCSRn are written to PMCn.PMCn_m.

Even when pins being used by multiple programs belong to the same port group, the PMCSRn register allows masking of the unused bits when overwriting so that independent setting of the corresponding bits by each program is possible.

Access: This register is readable/writable in 32-bit units.
Bits 31 to 16 are always read as 0000_H. Reading bits 15 to 0 returns the value of the PMCn register.

Value after reset: See **Section 2.2.1.1, List of the E1M-S Port Registers**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PMCSRn_31	PMCSRn_30	PMCSRn_29	PMCSRn_28	PMCSRn_27	PMCSRn_26	PMCSRn_25	PMCSRn_24	PMCSRn_23	PMCSRn_22	PMCSRn_21	PMCSRn_20	PMCSRn_19	PMCSRn_18	PMCSRn_17	PMCSRn_16
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PMCSRn_15	PMCSRn_14	PMCSRn_13	PMCSRn_12	PMCSRn_11	PMCSRn_10	PMCSRn_9	PMCSRn_8	PMCSRn_7	PMCSRn_6	PMCSRn_5	PMCSRn_4	PMCSRn_3	PMCSRn_2	PMCSRn_1	PMCSRn_0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 2.8 PMCSRn Register Contents

Bit Position	Bit Name	Function
31 to 16	PMCSRn_ [31:16]	Specify whether the value of the corresponding lower bit of PMCSRn_m is written to PMCn_m: 0: PMCn_m does not depend on PMCSRn_m. 1: The value of PMCn_m is the same as that of PMCSRn_m. Example: When PMCSRn.PMCSRn_31 = 1, the value of bit PMCSRn.PMCSRn_15 is written to bit PMCn.PMCn_15 and output.
15 to 0	PMCSRn_ [15:0]	Specify the PMCn_m value when the corresponding upper bit PMCSRn_(m+16) is 1: 0: PMCn_m = 0 1: PMCn_m = 1

(3) PIPCn — Port IP Control Register

This register specifies whether the I/O direction of pin Pn_m is controlled by the port mode register PMn.PMn_m or by an alternative function.

When the Pn_m pin is operated in alternative mode (PMCn.PMCn_m = 1) and the alternative function directly controls the I/O direction of Pn_m, PIPCn.PIPCn_m must be set to 1 as well. This hands over I/O control to the alternative function and overrules the PMn.PMn_m setting.

Access: This register is readable/writable in 16-bit units.

Value after reset: See **Section 2.2.1.1, List of the E1M-S Port Registers**.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PIPCn_15	PIPCn_14	PIPCn_13	PIPCn_12	PIPCn_11	PIPCn_10	PIPCn_9	PIPCn_8	PIPCn_7	PIPCn_6	PIPCn_5	PIPCn_4	PIPCn_3	PIPCn_2	PIPCn_1	PIPCn_0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 2.9 PIPCn Register Contents

Bit Position	Bit Name	Function
15 to 0	PIPCn_[15:0]	Specify the I/O control mode: 0: I/O mode is selected by PMn.PMn_m (S/W I/O control). 1: I/O mode is selected by peripheral function (direct I/O control).

(4) PMn — Port Mode Register

The PMn register specifies whether the individual pins of port group n are in input mode or in output mode.

Access: This register is readable/writable in 16-bit units.

Value after reset: See **Section 2.2.1.1, List of the E1M-S Port Registers**.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PMn_15	PMn_14	PMn_13	PMn_12	PMn_11	PMn_10	PMn_9	PMn_8	PMn_7	PMn_6	PMn_5	PMn_4	PMn_3	PMn_2	PMn_1	PMn_0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 2.10 PMn Register Contents

Bit Position	Bit Name	Function
15 to 0	PMn_[15:0]	Specify input/output mode of the corresponding pin: 0: Output mode (output enabled) 1: Input mode (output disabled)

NOTES

1. To use a pin in input port mode (PMCn.PMCn_m = 0 and PMn.PMn_m = 1), the input buffer must be enabled (PIBCn.PIBCn_m = 1).
2. PMn_m specifies the I/O direction in port mode (PMCn.PMCn_m = 0) and alternative mode (PMCn.PMCn_m=1) because PIPCn.PIPCn_m = 0 after reset.

(5) PMSRn — Port Mode Set/Reset Register

This register provides an alternative method to set a bit in the PMn register.

The 16 higher-order bits of PMSRn specify whether the data in the 16 lower-order bits of PMSRn are written to PMn.PMn_m.

Even when pins being used by multiple programs belong to the same port group, the PMSRn register allows masking of the unused bits when overwriting so that independent setting of the corresponding bits by each program is possible.

Access: This register is readable/writable in 32-bit units.
Bits 31 to 16 are always read as 0000_H. Reading bits 15 to 0 returns the value of the PMn register.

Value after reset: See **Section 2.2.1.1, List of the E1M-S Port Registers**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PMSRn_31	PMSRn_30	PMSRn_29	PMSRn_28	PMSRn_27	PMSRn_26	PMSRn_25	PMSRn_24	PMSRn_23	PMSRn_22	PMSRn_21	PMSRn_20	PMSRn_19	PMSRn_18	PMSRn_17	PMSRn_16
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PMSRn_15	PMSRn_14	PMSRn_13	PMSRn_12	PMSRn_11	PMSRn_10	PMSRn_9	PMSRn_8	PMSRn_7	PMSRn_6	PMSRn_5	PMSRn_4	PMSRn_3	PMSRn_2	PMSRn_1	PMSRn_0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 2.11 PMSRn Register Contents

Bit Position	Bit Name	Function
31 to 16	PMSRn_[31:16]	Specify whether the value of the corresponding lower bit of PMSRn_m is written to PMn_m: 0: PMn_m does not depend on PMSRn_m. 1: The value of PMn_m is the same as that of PMSRn_m. Example: When PMSRn.PMSRn_31 = 1, the value of bit PMSRn.PMSRn_15 is written to bit PMn.PMn_15 and output.
15 to 0	PMSRn_[15:0]	Specify the PMn_m value when the corresponding upper bit PMSRn_(m+16) is 1: 0: PMn_m = 0 1: PMn_m = 1

(6) PIBCn — Port Input Buffer Control Register

When a pin is being used in input port mode (PMCn.PMCn_m = 0 and PMn.PMn_m = 1), this register enables or disables the input buffer. However, when the pin is used as an input pin in S/W I/O control alternative mode (PMCn.PMCn_m = 1 and PIPCn.PIPCn_m = 0) or direct I/O control alternative mode (PMCn.PMCn_m = 1 and PIPCn.PIPCn_m = 1), set PIBCn.PIBCn_m = 0.

And when pins are in the both-direction mode(PBDCn.PBDCn_m-1), it is capable of selecting shared output level loop-back function and pin output level-read function with the setting of PIBCn.PIBCn_m.

Refer to **(1) PBDCn — Port Bidirectional Control Register** in **Section 2.1.4.3, Pin Data Input/Output**.

Access: This register is readable/writable in 16-bit units.

Value after reset: See **Section 2.2.1.1, List of the E1M-S Port Registers**.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PIBCn_15	PIBCn_14	PIBCn_13	PIBCn_12	PIBCn_11	PIBCn_10	PIBCn_9	PIBCn_8	PIBCn_7	PIBCn_6	PIBCn_5	PIBCn_4	PIBCn_3	PIBCn_2	PIBCn_1	PIBCn_0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 2.12 PIBCn Register Contents

Bit Position	Bit Name	Function
15 to 0	PIBCn_[15:0]	These bits are used to enable and disable the input buffer in input port mode. 0: Input buffer is disabled. 1: Input buffer is enabled.

NOTE

When the input buffer is disabled, a shoot-through current does not flow even if the pin level is in Hi-Z state. Thus the pin does not need to be fixed to a high or low level externally.

(7) PFCn — Port Function Control Register

This register, together with the PFCEn and PFCAEn registers, specifies an alternative function of the pins.

Some alternative functions directly controls input/output of pin Pn_m. For such an alternative functions PIPCn.PIPCn_m must be set to 1 as well.

For other alternative functions, input/output must be specified by PMn.PMn_m.

Access: This register is readable/writable in 16-bit units.

Value after reset: See **Section 2.2.1.1, List of the E1M-S Port Registers**.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PFCn_15	PFCn_14	PFCn_13	PFCn_12	PFCn_11	PFCn_10	PFCn_9	PFCn_8	PFCn_7	PFCn_6	PFCn_5	PFCn_4	PFCn_3	PFCn_2	PFCn_1	PFCn_0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 2.13 PFCn Register Contents

Bit Position	Bit Name	Function
15 to 0	PFCn_[15:0]	Specify an alternative function of a pin. See **Table 2.4, Outline of Alternative Mode Selection (PMCn.PMCn_m = 1)** for details.

(8) PFCEn — Port Function Control Expansion Register

This register, together with the PFCn and PFCAEn registers, specifies an alternative function of the pins.

Some alternative functions directly controls input/output of pin Pn_m. For such an alternative functions PIPCn.PIPCn_m must be set to 1 as well.

For other alternative functions, input/output must be specified by PMn.PMn_m.

Access: This register is readable/writable in 16-bit units.

Value after reset: See **Section 2.2.1.1, List of the E1M-S Port Registers**.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PFCEn_15	PFCEn_14	PFCEn_13	PFCEn_12	PFCEn_11	PFCEn_10	PFCEn_9	PFCEn_8	PFCEn_7	PFCEn_6	PFCEn_5	PFCEn_4	PFCEn_3	PFCEn_2	PFCEn_1	PFCEn_0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 2.14 PFCEn Register Contents

Bit Position	Bit Name	Function
15 to 0	PFCEn_[15:0]	Specify an alternative function of a pin. See **Table 2.4, Outline of Alternative Mode Selection (PMCn.PMCn_m = 1)** for details.

(9) PFCAEn — Port Function Control Additional Expansion Register

This register, together with the PFCn and PFCEn registers, specifies an alternative function of the pins.

Some alternative functions directly controls input/output of pin Pn_m. For such an alternative functions PIPCn.PIPCn_m must be set to 1 as well.

For other alternative functions, input/output must be specified by PMn.PMn_m.

Access: This register is readable/writable in 16-bit units.

Value after reset: See **Section 2.2.1.1, List of the E1M-S Port Registers**.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PFCAEn_15	PFCAEn_14	PFCAEn_13	PFCAEn_12	PFCAEn_11	PFCAEn_10	PFCAEn_9	PFCAEn_8	PFCAEn_7	PFCAEn_6	PFCAEn_5	PFCAEn_4	PFCAEn_3	PFCAEn_2	PFCAEn_1	PFCAEn_0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 2.15 PFCAEn Register Contents

Bit Position	Bit Name	Function
15 to 0	PFCAEn_[15:0]	Specify an alternative function of a pin. See **Table 2.4, Outline of Alternative Mode Selection (PMCn.PMCn_m = 1)** for details.

2.1.4.3 Pin Data Input/Output

(1) PBDCn — Port Bidirectional Control Register

This register enables the input buffer when a pin is used in output mode, and permits or prohibits bidirectional mode. The Pn_m pin level is read via PPRn.PPRn_m in bidirectional mode.

- Alternative output level loopback function
  When the Pn_m pin is used as the alternative output function, the actual pin output level based on the alternative output function can be looped back to the alternative input side by setting PBDCn.PBDCn_m = 1 and PIBCn.PIBCn_m= 0. For example, the pin output level based on the first alternative function can be looped back to the same alternative input side. Also the pin output level can be read via PPRn.PPRn_m.
- Pin output level read function
  When the Pn_m pin is used as the general output port function or the alternative output function, the actual pin output level can be read via PPRn.PPRn_m by setting PBDCn.PBDCn_m = 1 and PIBCn.PIBCn_m = 1. Under this setting, the pin output level will never be looped back to the alternative input side even in alternative output mode.

Access: This register is readable/writable in 16-bit units.

Value after reset: See **Section 2.2.1.1, List of the E1M-S Port Registers**.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PBDCn_15	PBDCn_14	PBDCn_13	PBDCn_12	PBDCn_11	PBDCn_10	PBDCn_9	PBDCn_8	PBDCn_7	PBDCn_6	PBDCn_5	PBDCn_4	PBDCn_3	PBDCn_2	PBDCn_1	PBDCn_0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 2.16 PBDCn Register Contents

Bit Position	Bit Name	Function
15 to 0	PBDCn_[15:0]	Enable/disable bidirectional mode of the corresponding pin: 0: Bidirectional mode is disabled. 1: Bidirectional mode is enabled.

(2) PPRn — Port Pin Read Register

This register reflects an actual Pn_m pin level, a Pn.Pn_m bit value, or an output level of the alternative function. The value to be read depends on various control settings as described in **Table 2.5, PPRn_m Read Values**.

Access: This register is only readable in 16-bit units.

Value after reset: See **Section 2.2.1.1, List of the E1M-S Port Registers**.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PPRn_15	PPRn_14	PPRn_13	PPRn_12	PPRn_11	PPRn_10	PPRn_9	PPRn_8	PPRn_7	PPRn_6	PPRn_5	PPRn_4	PPRn_3	PPRn_2	PPRn_1	PPRn_0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 2.17 PPRn Register Contents

Bit Position	Bit Name	Function
15 to 0	PPRn_[15:0]	Indicate a Pn_m pin level, a Pn.Pn_m value, or alternative function output level.

(3) Pn — Port Register

This register sets and holds the Pn.Pn_m data to be output via the related Pn_m port in output port mode (PMCn.PMCn_m = 0 and PMn.PMn_m = 0).

Access: This register is readable/writable in 16-bit units.

Value after reset: See **Section 2.2.1.1, List of the E1M-S Port Registers**.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Pn_15	Pn_14	Pn_13	Pn_12	Pn_11	Pn_10	Pn_9	Pn_8	Pn_7	Pn_6	Pn_5	Pn_4	Pn_3	Pn_2	Pn_1	Pn_0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 2.18 Pn Register Contents

Bit Position	Bit Name	Function
15 to 0	Pn_[15:0]	Set the output level of pin m (m = 0 to 15): 0: Low level output 1: High level output

NOTE

The bits of this register can be manipulated by various means; refer to the subsection, Write to the Pn Register in **Section 2.1.2.3, Pin Data Input/Output**.

(4) PNOTn — Port NOT Register

This register enables inverting the Pn_m bit of the port register without directly writing to Pn.

Access: This register is writable in 16-bit units. The read value is always 0000_H.

Value after reset: See **Section 2.2.1.1, List of the E1M-S Port Registers**.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PNOTn_15	PNOTn_14	PNOTn_13	PNOTn_12	PNOTn_11	PNOTn_10	PNOTn_9	PNOTn_8	PNOTn_7	PNOTn_6	PNOTn_5	PNOTn_4	PNOTn_3	PNOTn_2	PNOTn_1	PNOTn_0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Table 2.19 PNOTn Register Contents

Bit Position	Bit Name	Function
15 to 0	PNOTn_[15:0]	Specify if Pn.Pn_m is inverted or not: 0: Pn.Pn_m is not inverted (Pn_m → Pn_m). 1: Pn.Pn_m is inverted ($\overline{Pn\_m}$ → Pn_m)

(5) PSRn — Port Set/Reset Register

This register provides an alternative method to set a bit in the Pn register.

The 16 higher-order bits of PSRn specify whether the data in the 16 lower-order bits of PSRn are written to Pn.Pn_m.

Even when pins being used by multiple programs belong to the same port group, the PSRn register allows masking of the unused bits when overwriting so that independent setting of the corresponding bits by each program is possible.

Access: This register is readable/writable in 32-bit units.
Bits 31 to 16 are always read as 0000_H. Reading bits 15 to 0 returns the value of the PMCn register.

Value after reset: See **Section 2.2.1.1, List of the E1M-S Port Registers**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PSRn_31	PSRn_30	PSRn_29	PSRn_28	PSRn_27	PSRn_26	PSRn_25	PSRn_24	PSRn_23	PSRn_22	PSRn_21	PSRn_20	PSRn_19	PSRn_18	PSRn_17	PSRn_16
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PSRn_15	PSRn_14	PSRn_13	PSRn_12	PSRn_11	PSRn_10	PSRn_9	PSRn_8	PSRn_7	PSRn_6	PSRn_5	PSRn_4	PSRn_3	PSRn_2	PSRn_1	PSRn_0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 2.20 PSRn Register Contents

Bit Position	Bit Name	Function
31 to 16	PSRn_[31:16]	Specify whether the value of the corresponding lower bit of PSRn_m is written to Pn_m: 0: Pn_m does not depend on PSRn_m. 1: The value of Pn_m is the same as that of PSRn_m. Example: When PSRn.PSRn_31 = 1, the value of bit PSRn.PSRn_15 is written to bit Pn.Pn_15 and output.
15 to 0	PSRn_[15:0]	Specify the Pn_m value when the corresponding upper bit PSRn_(m+16) is 1: 0: Pn_m = 0 1: Pn_m = 1

(6) PINVn — Port Output Level Inversion Register

This register enables inverting the output level from a pin. It is effective when the pin is in output mode regardless of port output mode or alternative output mode.

Access: This register is readable/writable in 32-bit units.
Updating of this register needs a correct write sequence using the PPCMDn register.

Value after reset: See **Section 2.2.1.1, List of the E1M-S Port Registers**.

CAUTION

The upper 16 bits are also intended for the inverted value of a port register write sequence that is to be protected. See Section 2.1.4.5, Port Register Protection for details.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PINVn_15	PINVn_14	PINVn_13	PINVn_12	PINVn_11	PINVn_10	PINVn_9	PINVn_8	PINVn_7	PINVn_6	PINVn_5	PINVn_4	PINVn_3	PINVn_2	PINVn_1	PINVn_0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 2.21 PINVn Register Contents

Bit Position	Bit Name	Function
15 to 0	PINVn_[15:0]	Specify whether the output level from a pin is inverted or not. 0: Pin output level is not inverted. 1: Pin output level is inverted.

2.1.4.4 Configuration of Electrical Characteristics

(1) PUn — Pull-Up Option Register

This register specifies whether an on-chip pull-up resistor is connected to an input pin.

Access: This register is readable/writable in 16-bit units.

Value after reset: See **Section 2.2.1.1, List of the E1M-S Port Registers**.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PUn_15	PUn_14	PUn_13	PUn_12	PUn_11	PUn_10	PUn_9	PUn_8	PUn_7	PUn_6	PUn_5	PUn_4	PUn_3	PUn_2	PUn_1	PUn_0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 2.22 PUn Register Contents

Bit Position	Bit Name	Function
15 to 0	PUn_[15:0]	Specify whether an on-chip pull-up resistor is connected to the corresponding pin: 0: No on-chip pull-up resistor is connected. 1: On-chip pull-up resistor is connected.

NOTES

1. Do not set PUn.PUn_m =1 and PDn.PDn_m =1 to a single pin.
2. The on-chip pull-up resistor has no effect when the pin is operated in output mode.

(2) PDn — Pull-Down Option Register

This register specifies whether an on-chip pull-down resistor is connected to an input pin.

Access: This register is readable/writable in 16-bit units.

Value after reset: See **Section 2.2.1.1, List of the E1M-S Port Registers**.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PDn_15	PDn_14	PDn_13	PDn_12	PDn_11	PDn_10	PDn_9	PDn_8	PDn_7	PDn_6	PDn_5	PDn_4	PDn_3	PDn_2	PDn_1	PDn_0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 2.23 PDn Register Contents

Bit Position	Bit Name	Function
15 to 0	PDn_[15:0]	Specify whether an on-chip pull-down resistor is connected to the corresponding pin: 0: No on-chip pull-down resistor is connected. 1: On-chip pull-down resistor is connected.

NOTES

1. Do not set PUn.PUn_m =1 and PDn.PDn_m =1 to a single pin.
2. The on-chip pull-down resistor has no effect when the pin is operated in output mode.

(3) PDSCn — Port Drive Strength Control Register

This register is used to configure driving ability (driving strength) of output port pins and multiplex output pin functions in combination with the PUCCn register. Specific driving ability settings may be required, depending on the pin function to be used. For details, see **Section 37 Electrical Characteristics**.

Access: This register is readable/writable in 32-bit units.
Updating of this register needs a correct write sequence using the PPCMDn register.

Value after reset: See **Section 2.2.1.1, List of the E1M-S Port Registers**.

CAUTION

For protected port registers, the inverse of the value of the 16 lower-order bits is written to the 16 higher-order bits as part of the sequence for writing to a protected register. See Section 2.1.4.5, Port Register Protection for details.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PDSCn_15	PDSCn_14	PDSCn_13	PDSCn_12	PDSCn_11	PDSCn_10	PDSCn_9	PDSCn_8	PDSCn_7	PDSCn_6	PDSCn_5	PDSCn_4	PDSCn_3	PDSCn_2	PDSCn_1	PDSCn_0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 2.24 PDSCn Register Contents

Bit Position	Bit Name	Function
15 to 0	PDSCn_[15:0]	Set driving ability of output port pins and multiplex output pin functions in combination with the PUCCn register. See **Table 2.25 PUCCn Register Contents** for register setting and output driving ability.

(4) PUCCn — Port Universal Control Register

This register is used to configure driving ability (driving strength) of output port pins and multiplex output pin functions in combination with the PDSCn register. Specific driving ability settings may be required, depending on the pin function to be used. For details, see **Section 37, Electrical Characteristics**.

Access: This register is readable/writable in 32-bit units.
Updating of this register needs a correct write sequence using the PPCMDn register.

Value after reset: See **Section 2.2.1.1, List of the E1M-S Port Registers**.

CAUTION

For protected port registers, the inverse of the value of the 16 lower-order bits is written to the 16 higher-order bits as part of the sequence for writing to a protected register. See Section 2.1.4.5, Port Register Protection for details.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PUCCn_15	PUCCn_14	PUCCn_13	PUCCn_12	PUCCn_11	PUCCn_10	PUCCn_9	PUCCn_8	PUCCn_7	PUCCn_6	PUCCn_5	PUCCn_4	PUCCn_3	PUCCn_2	PUCCn_1	PUCCn_0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 2.25 PUCCn Register Contents

Bit Position	Bit Name	Function
15 to 0	PUCCn_[15:0]	Set driving ability of output port pins and multiplex output pin functions in combination with the PDSCn register. For electrical characteristics for each setting, see **Section 37.3.4, Output Slew Rate**.

PUCCn_m	PDSCn_m	Output Driving Ability
0	0	Low
0	1	High
1	0	Mid
1	1	Setting is prohibited.

2.1.4.5 Port Register Protection

(1) PPCMDn — Port Register Protection Command Register

This register is a command register for the port register to be protected.

Access: This register is writable in 8-bit units.
The read value of bit 7 to 0 is always 0.

Value after reset: See **Section 2.2.1.1, List of the E1M-S Port Registers**.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—
R/W	W	W	W	W	W	W	W	W

Table 2.26 PPCMDn Register Contents

Bit Position	Bit Name	Function
7 to 0	—	The protection unlock sequence is entered by writing the fixed value ($A5_H$).

(2) PPROTSn — Port Protection Status Register

This register indicates the status of a port register write sequence to be protected.

Access: This register is readable in 8-bit units.
Writing to this register is ignored.

Value after reset: See **Section 2.2.1.1, List of the E1M-S Port Registers**.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	PPROTSn_0
R/W	R	R	R	R	R	R	R	R

Table 2.27 PPROTSn Register Contents

Bit Position	Bit Name	Function
0	PPROTSn_0	Checks a write sequence error of the port register to be protected. 0: Protection error does not occur. 1: Protection error occurs.

(3) Port Register to be Protected

- Port Drive Strength Control Register (PDSCn)
- Port Universal Control Register (PUCCn)
- Port Output Level Inversion Register (PINVn)

(4) Sequence of Writing to Port Registers for Protection

Writing to the write-protected registers is only possible with the protection unlock sequence described below.

Procedure

1) Write the fixed value ($A5_H$) to the port register protection command register PPCMDn.
2) Write the new setting to the write-protected register. Write the value after a reset to the reserved bits.
3) Write the bitwise inverse of the setting value to the same register as in step 2). Write the inverse of the value after a reset to the reserved bits.
4) Write the new setting to the same register as in step 2). Write the value after a reset to the reserved bits.

A value can only be written to a write-protected register by following the procedure above.

Protection is not unlocked if the procedure is not followed correctly, the setting is not written to the write-protected register, and the PPROTSn_0 bit of the port protection status register PPROTSn is set to 1. (Although this is not a requirement, you can confirm if the values set in the targeted register was correctly written by checking that the setting of the PPROTSn_0 bit is 0 after step 4).)

In case of failure to follow the sequence, repeat the procedure from the beginning.

In case of writing to another register during steps 1) to 4) of the sequence, the protection function operates as follows. Operation is also as described when any interrupt is accepted during the sequence and the interrupt handling includes access to another register.

- When the writing is to another register in the same module*1, writing to the protected register fails and the PPROTSn_0 bit is set to 1.
- When writing is to a register in another module, writing to the protected register will successfully proceed to completion, if the remainder of the procedure is followed correctly.

Reading of another register while the sequence is in progress does not lead to failure.

Note 1. The port group configuration registers are divided into two modules. Each module includes the full set of port configuration registers, including protection-related registers.

- Port group 0
- Port groups other than port group 0

2.1.4.6 Pin-Unit Register

(1) PCRn_m — Port Control Register

Each register of a port group can be accessed via this register and a PCRn_m register can set all functions of a single pin. For example, setting bit 6 of the PCRn_m register to 1 sets bit m of the PMCn register to 1 also. Although writing to the PDSCn, PUCCn, and PINVn registers are protected, an access to them via the PCRn_m register does not require a protection unlock sequence.

CAUTIONS

- **Multiple bits in the PCRn_m register can be batch-set within the ranges described in Section 2.1.4.7, Example of Port Configuration Flow (2), Individual Setting.**
- **If the bits are batch-set out of the range for setting of PCRn_m, the pin may output an unexpected signal level.**

Access: This register is readable/writable in 32-bit units.

Value after reset: See **Section 2.2.1.1, List of the E1M-S Port Registers**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	PINV	—	—	—	—	PUCC	PDSC	—	—	—	—	PU	PD	PBDC	PIBC
R/W	R	R/W	R	R	R	R	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	P	—	—	—	PPR	—	PMC	PIPC	PM	—	PFCAE	PFCE	PFC
R/W	R	R	R	R/W	R	R	R	R	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Table 2.28 PCRn_m Register Contents

Bit Position	Bit Name	Function
31	—	Reserved. The read value is always 0. The write value should always be 0.
30	PINV	Same function as bit m of the PINVn register
29 to 26	—	Reserved. The read values are always 0. The write values should always be 0.
25	PUCC	Same function as bit m of the PUCCn register
24	PDSC	Same function as bit m of the PDSCn register
23 to 20	—	Reserved. The read values are always 0. The write values should always be 0.
19	PU	Same function as bit m of the PUn register
18	PD	Same function as bit m of the PDn register
17	PBDC	Same function as bit m of the PBDCn register
16	PIBC	Same function as bit m of the PIBCn register
15 to 13	—	Reserved. The read values are always 0. The write values should always be 0.
12	P	Same function as bit m of the Pn register
11 to 9	—	Reserved. The read values are always 0. The write values should always be 0.
8	PPR	Same function as bit m of the PPRn register
7	—	Reserved. The read value is always 0. The write value should always be 0.
6	PMC	Same function as bit m of the PMCn register
5	PIPC	Same function as bit m of the PIPCn register
4	PM	Same function as bit m of the PMn register
3	—	Reserved. The read value is always 0. The write value should always be 0.

Table 2.28 PCRn_m Register Contents

Bit Position	Bit Name	Function
2	PFCAE	Same function as bit m of the PFCAEn register
1	PFCE	Same function as bit m of the PFCEn register
0	PFC	Same function as bit m of the PFCn register

2.1.4.7 Example of Port Configuration Flow

The followings are examples of the port configuration flow. For port filter setting of each flow chart, see **Section 2.3.2.4, Setting Procedures of Peripheral Function DNF** or **Section 2.3.3.6, Setting Procedures of Edge Detection DNF**.

CAUTION

Even if a port is used in S/W I/O control alternative mode, it may be temporarily switched to alternative input mode in the following port configuration flows. This may occur during the period from setting of PMCn_m = 1 to that of PMn_m = 0.

(1) Batch Setting

The following shows an example to collectively set a port group.

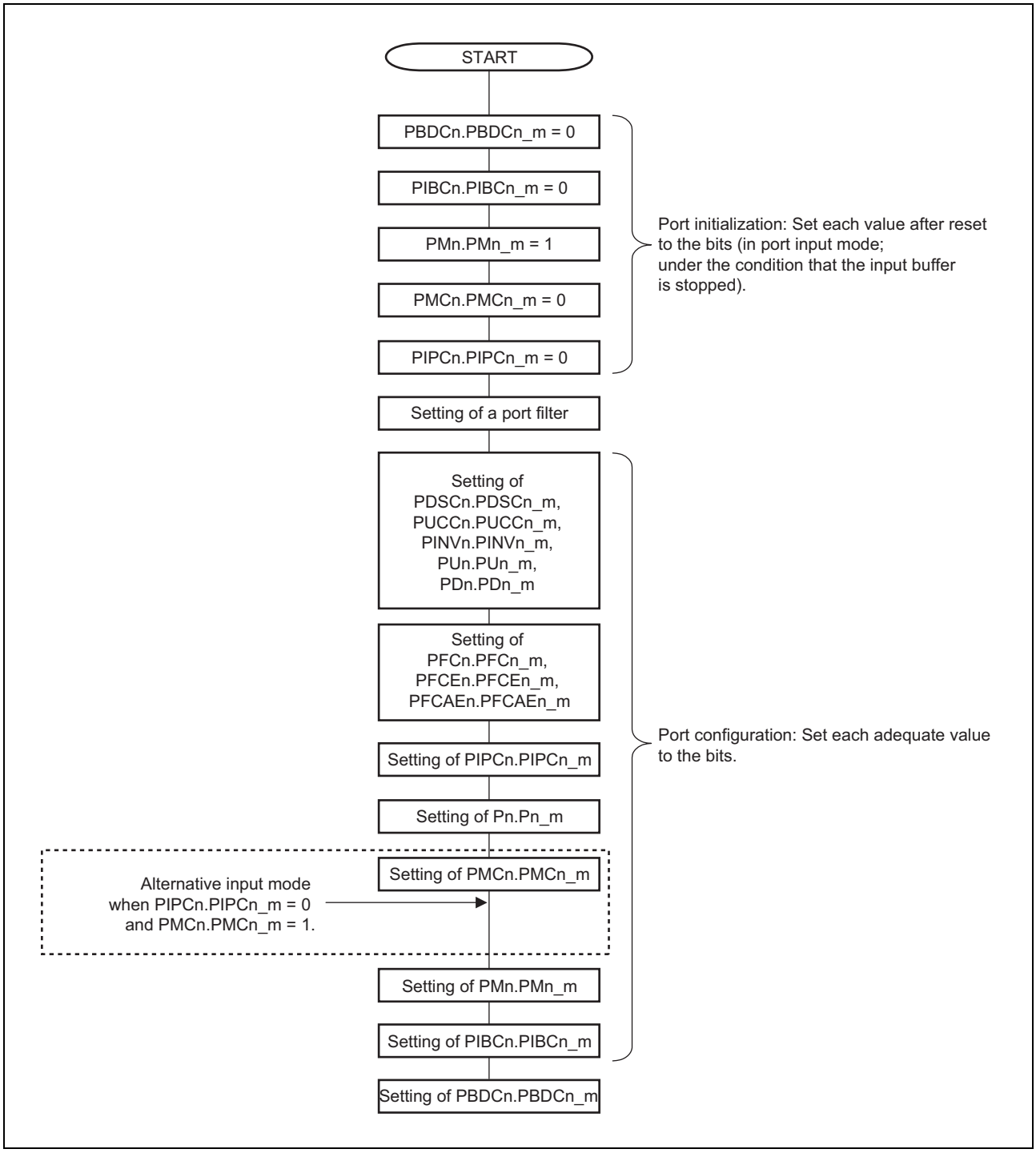


Figure 2.3 Example of Port Configuration Flow (Batch Setting)

(2) Individual Setting

The following shows an example to individually set a port group.

And it is capable of setting simultaneous bits at once within the Settable range described below by using PCRn_m register.

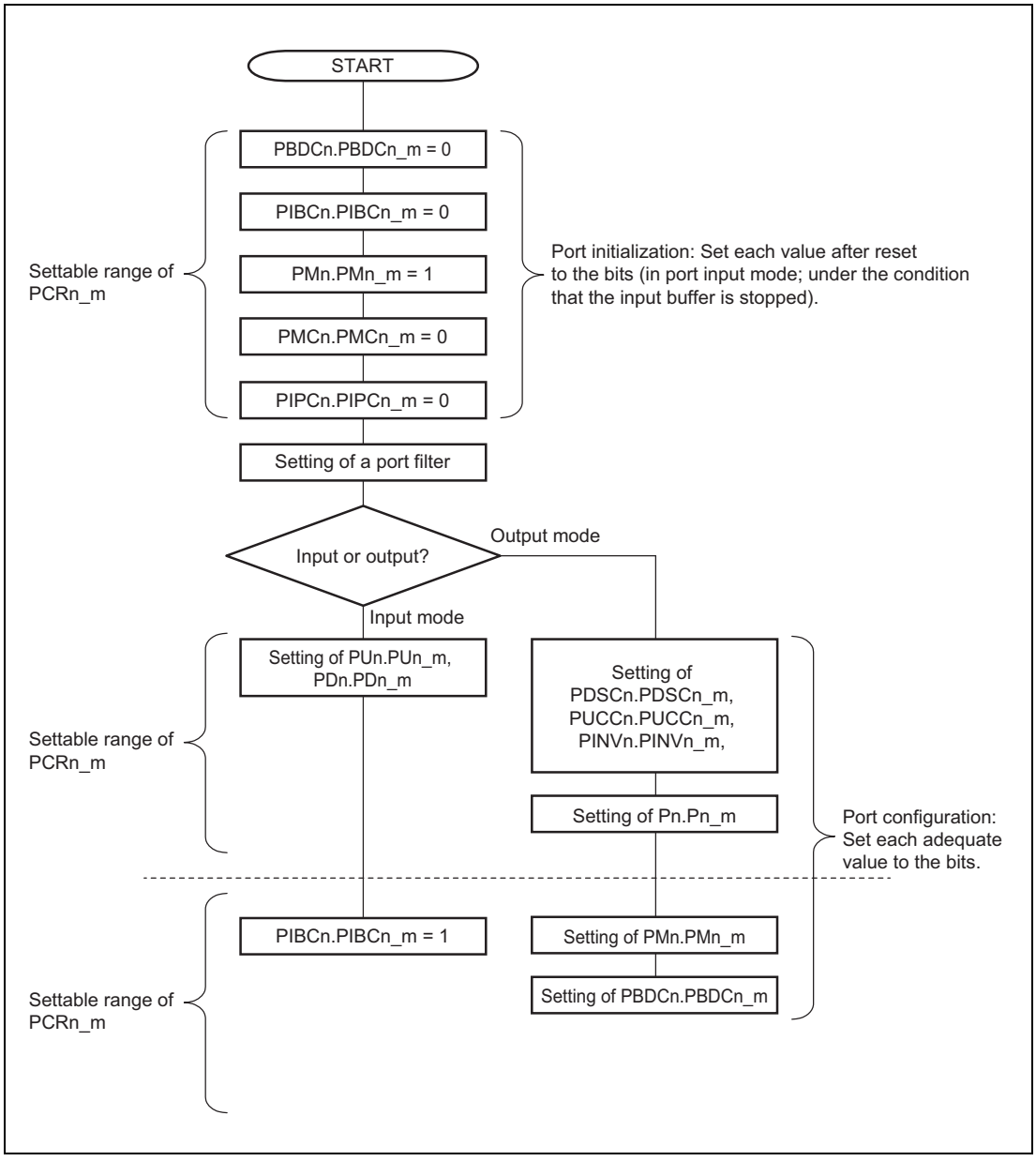


Figure 2.4 Example of Port Configuration Flow (in Port Mode)

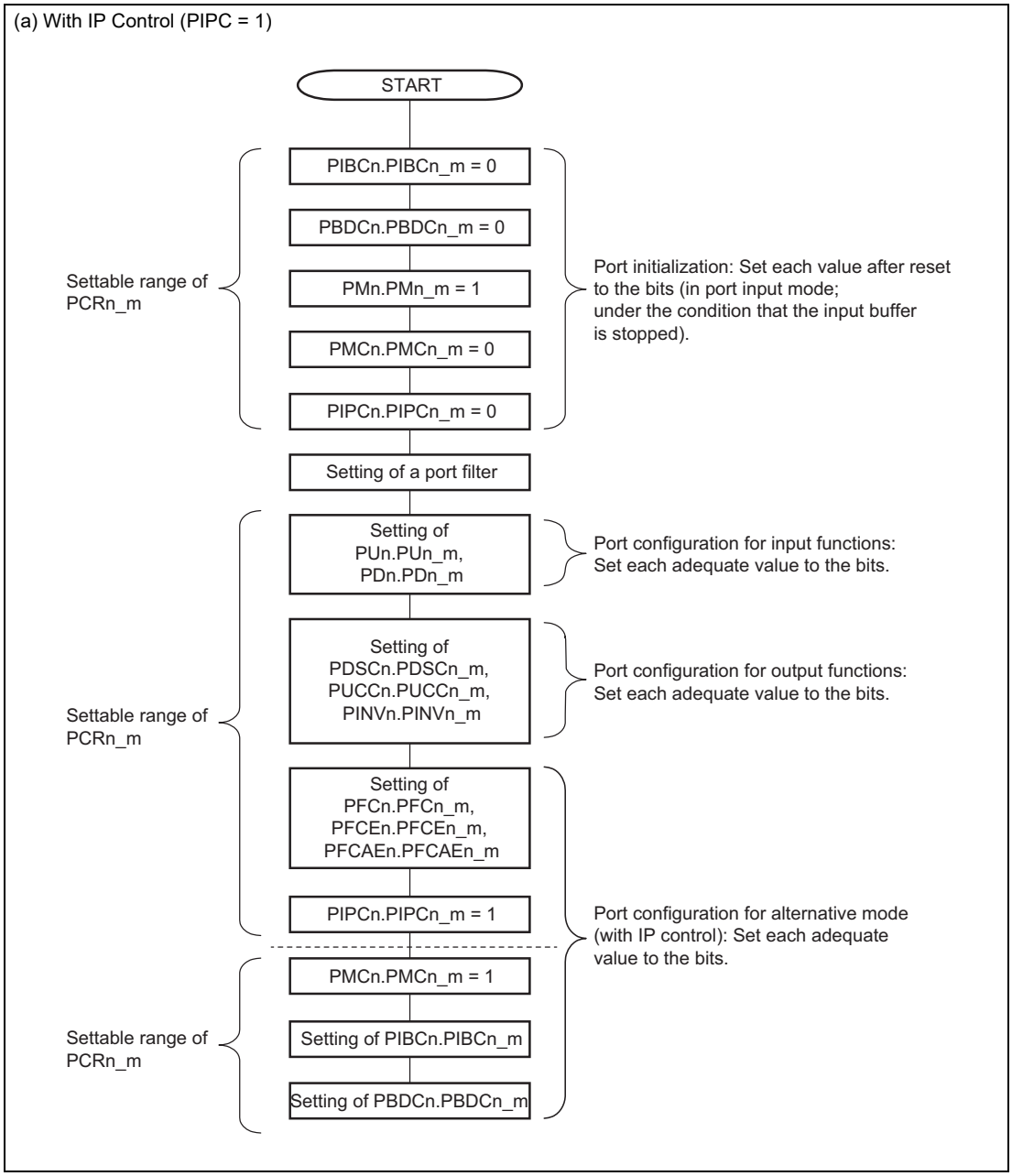


Figure 2.5 Example of Port Configuration Flow (in Alternative Mode)(1/2)

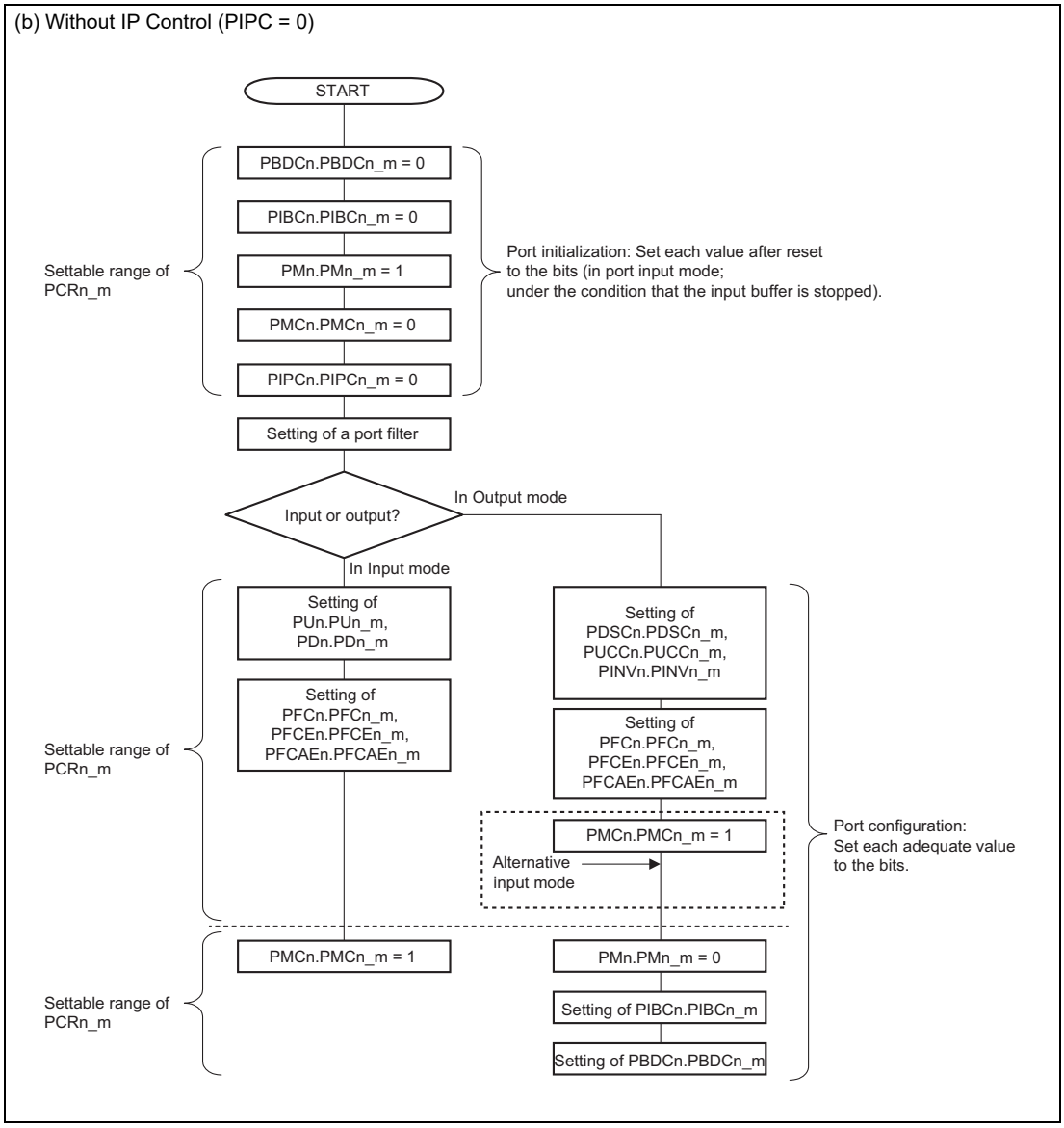


Figure 2.6 Example of Port Configuration Flow (in Alternative Mode) (2/2)

2.1.5 Functional Selection

2.1.5.1 Register Configuration in Use of the Alternative Function

When the pin alternative function is used, set PMCn_m = 1 and select the alternative numbers of PFCn_m, PFCEn_m, and PFCAEn_m. In several peripheral functions, a single alternative I/O function is allocated to multiple pins. However, such an alternative function should not be enabled in multiple pins at the same time. For example, an external interrupt input, IRQ0 is allocated to P0_3 and P2_0, but this alternative function can only be selected in either of them.

2.1.5.2 Alternative Function to be used in Direct I/O Control Alternative Mode

When the alternative functions described in **Table 2.29** are used, switch to direct I/O control alternative mode. When setting PIPCn_m = 1, the PMn_m value which has been set is ignored because the peripheral function enables or disables inputs and outputs of the buffer.

In addition, set PIPCn_m = 0 if you are using an alternative function not listed in **Table 2.29**.

Table 2.29 List of the Pins which Require PIPC Register Setting

Category	Pin Name	I/O	Pin Function	Input/Output Control by Peripheral Functions
SCIn	SCInRxD	I	Receive data input	Input through the pin is enabled when the SCI3nSCR.RE bit is enabling reception by SCI3.
	SCInTxD	O	Transmit data output	Output through the pin is enabled when the SCI3nSCR.TE bit is enabling transmission by SCI3.
	SCInSCK	I/O	Serial clock input/output	The SCI3nSCR.CKE[1:0] bits determine whether the pin is an input or output. • Asynchronous transfer CKE[1:0] = 00 : Both input and output through the pin are disabled. CKE[1:0] = 01 : The pin is an output. CKE[1:0] = 1x : Prohibited settings • Synchronous transfer CKE[1:0] = 0x : The pin is an output. CKE[1:0] = 1x : The pin is an input.
CSIHn	CSIHnTSO	O	Transmit data output (data integrity verification)	• Output control See **(3), Operation of CSIHTSSO** in **Section 13.5.9, Communication in Slave Mode**. • Input control When the CSIHnCTL1.DCS bit is enabling checking the consistency of data through CSIH, enabling the input buffer for the pin leads to the actual output level of the pin being looped back within the device.
	CSIHnTSCK	I/O	Serial clock input/output	When then setting of SCIHnCTL2 is for CSIH operation in slave mode, this pin is an input. When the setting is for master mode. it is an output.
	CSIHnTRY	I/O	Handshake signal input/output	When the setting of CSIHnCTL2 is for CSIH operation in slave mode, this pin is an output. When the setting is for master mode, it is an input.
TSG2n	TSOn1 to TSOn6	O	Timer PWM output (Hi-Z control)	This output pin is forcibly placed in the high-impedance state under TAPA control in the event of an abnormality. Output though the pin is enabled in normal operation.
ATU-IV Timer C	TIOCn0 to TIOCn3	I/O	Input capture trigger input, output compare output	This pin is an output when ATU-IV timer C is set for compare-match operation by the TIORCx register. This pin is an input when the setting is for input-capture operation.

Note: n indicates the number of a given peripheral module.

2.1.5.3 Setting of the $\overline{\text{ERROROUT\_C}}$ Pin

When an error output function of the $\overline{\text{ERROROUT\_C}}$ pin is used, enable the alternative output level loopback function for fault diagnosis.

2.1.5.4 Register Setting in Use of an Analog Input Pin

Since ANxxx(SAR-A/D) DSANxP and DSANxN(DS-A/D) for an analog input are always connected to the A/D converter, setting of a port register to select a pin function is not required. However, when an analog input pin, which is also used for a general input port, is used, set PMCn_m = 0*1 and PIBCn_m = 0 (the value after reset of the port register). This prevents shoot-through current of the input buffer on the digital side during an analog voltage input. The relevant pins of this product are port groups P14, P15, and P16.

Note 1. Some pins may not be equipped with a PMCn_m bit. For which pins are and are not thus equipped, see the list of registers in **Section 2.2, Organization of Port Groups**.

2.1.5.5 Register Setting in Use of the LVDS buffer

When the LVDS buffer of RHSB is to be used, set PM17_m = 0 to enable output. The output from the LVDS buffer actually being enabled takes at least 10 µs after setting PM17_m = 0.
The LVDS buffer and PM17_m mutually correspond as shown in the table below. " √ " indicates with a support function and " — " without. For the registers without support functions, the settings should not be modified (the settings should be fixed to their values after reset).

		E1M-S	
Port Register	**LVDS Buffer**	**BGA304**	**BGA252**
PM17_0	RHSB0SOP	√	√
	RHSB0SON		
PM17_1	RHSB0FCLP	√	√
	RHSB0FCLN		
PM17_2	RHSB1SOP	√	√
	RHSB1SON		
PM17_3	RHSB1FCLP	√	√
	RHSB1FCLN		

2.1.5.6 Selecting of Function for JTAG Port

For the connection of multiple tools to the JTAG port, the interface is multiplexed on multiple sets of pins. The interface is selected by the combination of the settings of mode pins and, if the mode pins are selecting user boot mode as the operating mode, option byte OPBT2.

When any other operating mode is selected, the setting of OPBT2 has no effect and the I/F that corresponds to the operating mode is selected. For details, refer to **Section 5.2, Operating Mode**.

2.2 Organization of Port Groups

2.2.1 E1M-S Port Function

2.2.1.1 List of the E1M-S Port Registers

Table 2.30 to **Table 2.45** show detailed bitmaps of control registers in each port group. In the bitmap field, " √ " means an effective bit and " — " means a reserved one. Reserved areas are always read as values after reset. The write value also should be a value after reset.

The pins of port groups 7, 8, and 10 are included in the BGA304 package, but not in BGA252. For the port registers of the pins not included, the setting should not be modified (the settings should be fixed to their values after reset).

Table 2.30 List of Registers in E1M-S Port Group 0

Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Remarks
0	P0	R/W	0000_H	16	—	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
	PSR0	R/W	$0000\ 0000_H$	32	—	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
					—	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
	PPR0	R	0000_H	16	—	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
	PM0	R/W	$FFFF_H$	16	—	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
	PMC0	R/W	0000_H	16	—	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
	PFC0	R/W	0000_H	16	—	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
	PFCE0	R/W	0000_H	16	—	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
	PFCAE0	R/W	0000_H	16	—	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
	PNOT0	W	0000_H	16	—	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
	PMSR0	R/W	$0000\ FFFF_H$	32	—	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
					—	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
	PMCSR0	R/W	$0000\ 0000_H$	32	—	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
					—	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
	PIBC0	R/W	0000_H	16	—	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
	PBDC0	R/W	0000_H	16	—	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
	PIPC0	R/W	0000_H	16	—	√	√	√	—	√	—	√	√	—	—	—	—	√	√	√	
	PU0	R/W	0000_H	16	—	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
	PD0	R/W	0000_H	16	—	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
	PDSC0	R/W	$0000\ 0000_H$	32	—	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
					—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	Upper 16 bits
	PUCC0	R/W	$0000\ 0000_H$	32	—	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
					—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	Upper 16 bits
	PPROTS0	R	00_H	8	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	√	
	PPCMD0	W	00_H	8	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
	PINV0	R/W	$0000\ 0000_H$	32	—	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
					—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	Upper 16 bits

Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap 30 PINV	25 PUCC	24 PDSC	19 PU	18 PD	17 PBDC	16 PIBC	12 P	8 PPR	6 PMC	5 PIPC	4 PM	2 PFCAE	1 PFCE	0 PFC
0	PCR0_0	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR0_1	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR0_2	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR0_3	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	—	√	√	√	√
	PCR0_4	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	—	√	√	√	√
	PCR0_5	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	—	√	√	√	√
	PCR0_6	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	—	√	√	√	√
	PCR0_7	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR0_8	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR0_9	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	—	√	√	√	√
	PCR0_10	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR0_11	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	—	√	√	√	√
	PCR0_12	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR0_13	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR0_14	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√

Table 2.31 List of Registers in E1M-S Port Group 1

Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Remarks
1	P1	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
	PSR1	R/W	$0000\ 0000_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
					√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
	PPR1	R	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
	PM1	R/W	$FFFF_H$	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
	PMC1	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
	PFC1	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
	PFCE1	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
	PFCAE1	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
	PNOT1	W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
	PMSR1	R/W	$0000\ FFFF_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
					√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
	PMCSR1	R/W	$0000\ 0000_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
					√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
	PIBC1	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
	PBDC1	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
	PIPC1	R/W	0000_H	16	√	√	√	√	√	√	—	—	√	√	√	√	√	√	√	√	
	PU1	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
	PD1	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
	PDSC1	R/W	$0000\ 0000_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
					—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	Upper 16 bits
	PUCC1	R/W	$0000\ 0000_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
					—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	Upper 16 bits
	PPROTS1	R	00_H	8	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	√	
	PPCMD1	W	00_H	8	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
	PINV1	R/W	$0000\ 0000_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
					—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	Upper 16 bits

Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap 30 PINV	25 PUCC	24 PDSC	19 PU	18 PD	17 PBDC	16 PIBC	12 P	8 PPR	6 PMC	5 PIPC	4 PM	2 PFCAE	1 PFCE	0 PFC
1	PCR1_0	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR1_1	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR1_2	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR1_3	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR1_4	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR1_5	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR1_6	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR1_7	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR1_8	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	—	√	√	√	√
	PCR1_9	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	—	√	√	√	√
	PCR1_10	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR1_11	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR1_12	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR1_13	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR1_14	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR1_15	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√

Table 2.32 List of Registers in E1M-S Port Group 2

Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Remarks
2	P2	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
	PSR2	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Lower 16 bits
					—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Upper 16 bits
	PPR2	R	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
	PM2	R/W	$FFFF_H$	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
	PMC2	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
	PFC2	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
	PFCE2	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
	PFCAE2	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
	PNOT2	W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
	PMSR2	R/W	$0000\ FFFF_H$	32	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Lower 16 bits
					—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Upper 16 bits
	PMCSR2	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Lower 16 bits
					—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Upper 16 bits
	PIBC2	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
	PBDC2	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
	PIPC2	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
	PU2	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
	PD2	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
	PDSC2	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Lower 16 bits
					—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	Upper 16 bits
	PUCC2	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Lower 16 bits
					—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	Upper 16 bits
	PPROTS2	R	00_H	8	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	√	
	PPCMD2	W	00_H	8	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
	PINV2	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Lower 16 bits
					—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	Upper 16 bits

Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap 30 PINV	25 PUCC	24 PDSC	19 PU	18 PD	17 PBDC	16 PIBC	12 P	8 PPR	6 PMC	5 PIPC	4 PM	2 PFCAE	1 PFCE	0 PFC
2	PCR2_0	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR2_1	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR2_2	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR2_3	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR2_4	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR2_5	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR2_6	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR2_7	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√

Table 2.33 List of Registers in E1M-S Port Group 3

Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Remarks
3	P3	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
	PSR3	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Lower 16 bits
					—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Upper 16 bits
	PPR3	R	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
	PM3	R/W	$FFFF_H$	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
	PMC3	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
	PFC3	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
	PFCE3	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
	PFCAE3	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
	PNOT3	W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
	PMSR3	R/W	$0000\ FFFF_H$	32	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Lower 16 bits
					—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Upper 16 bits
	PMCSR3	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Lower 16 bits
					—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Upper 16 bits
	PIBC3	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
	PBDC3	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
	PIPC3	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
	PU3	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
	PD3	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
	PDSC3	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Lower 16 bits
					—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	Upper 16 bits
	PUCC3	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Lower 16 bits
					—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	Upper 16 bits
	PPROTS3	R	00_H	8	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	√	
	PPCMD3	W	00_H	8	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
	PINV3	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Lower 16 bits
					—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	Upper 16 bits

Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap 30 PINV	25 PUCC	24 PDSC	19 PU	18 PD	17 PBDC	16 PIBC	12 P	8 PPR	6 PMC	5 PIPC	4 PM	2 PFCAE	1 PFCE	0 PFC
3	PCR3_0	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR3_1	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR3_2	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR3_3	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR3_4	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR3_5	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR3_6	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR3_7	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√

Table 2.34 List of Registers in E1M-S Port Group 4

Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Remarks
4	P4	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
	PSR4	R/W	$0000\ 0000_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
					√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
	PPR4	R	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
	PM4	R/W	$FFFF_H$	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
	PMC4	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
	PFC4	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
	PFCE4	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
	PFCAE4	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
	PNOT4	W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
	PMSR4	R/W	$0000\ FFFF_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
					√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
	PMCSR4	R/W	$0000\ 0000_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
					√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
	PIBC4	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
	PBDC4	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
	PIPC4	R/W	0000_H	16	—	—	√	√	—	—	√	√	—	—	—	√	√	√	√	√	
	PU4	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
	PD4	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
	PDSC4	R/W	$0000\ 0000_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
					—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	Upper 16 bits
	PUCC4	R/W	$0000\ 0000_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
					—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	Upper 16 bits
	PPROTS4	R	00_H	8	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	√	
	PPCMD4	W	00_H	8	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
	PINV4	R/W	$0000\ 0000_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
					—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	Upper 16 bits

Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap 30 PINV	25 PUCC	24 PDSC	19 PU	18 PD	17 PBDC	16 PIBC	12 P	8 PPR	6 PMC	5 PIPC	4 PM	2 PFCAE	1 PFCE	0 PFC
4	PCR4_0	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR4_1	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR4_2	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR4_3	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR4_4	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR4_5	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	—	√	√	√	√
	PCR4_6	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	—	√	√	√	√
	PCR4_7	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	—	√	√	√	√
	PCR4_8	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR4_9	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR4_10	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	—	√	√	√	√
	PCR4_11	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	—	√	√	√	√
	PCR4_12	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR4_13	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR4_14	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	—	√	√	√	√
	PCR4_15	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	—	√	√	√	√

Table 2.35 List of Registers in E1M-S Port Group 5

Port Group Name	Register name	R/W	Value after Reset	Access Size	Bitmap 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Remarks
5	P5	R/W	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
	PSR5	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
					—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
	PPR5	R	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
	PM5	R/W	$FFFF_H$	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
	PMC5	R/W	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
	PFC5	R/W	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
	PFCE5	R/W	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
	PFCAE5	R/W	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
	PNOT5	W	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
	PMSR5	R/W	$0000\ FFFF_H$	32	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
					—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
	PMCSR5	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
					—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
	PIBC5	R/W	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
	PBDC5	R/W	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
	PIPC5	R/W	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
	PU5	R/W	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
	PD5	R/W	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
	PDSC5	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
					—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	Upper 16 bits
	PUCC5	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
					—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	Upper 16 bits
	PPROTS5	R	00_H	8	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	√	
	PPCMD5	W	00_H	8	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
	PINV5	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
					—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	Upper 16 bits

Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap 30 PINV	25 PUCC	24 PDSC	19 PU	18 PD	17 PBDC	16 PIBC	12 P	8 PPR	6 PMC	5 PIPC	4 PM	2 PFCAE	1 PFCE	0 PFC
5	PCR5_0	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR5_1	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR5_2	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR5_3	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR5_4	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR5_5	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR5_6	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR5_7	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR5_8	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR5_9	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√

Table 2.36 List of Registers in E1M-S Port Group 6

Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Remarks
6	P6	R/W	0000_H	16	—	—	—	—	√	√	√	√	√	√	√	√	√	√	√	√	
	PSR6	R/W	$0000\ 0000_H$	32	—	—	—	—	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
					—	—	—	—	√	√	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
	PPR6	R	0000_H	16	—	—	—	—	√	√	√	√	√	√	√	√	√	√	√	√	
	PM6	R/W	$FFFF_H$	16	—	—	—	—	√	√	√	√	√	√	√	√	√	√	√	√	
	PMC6	R/W	0000_H	16	—	—	—	—	√	√	√	√	√	√	√	√	√	√	√	√	
	PFC6	R/W	0000_H	16	—	—	—	—	√	√	√	√	√	√	√	√	√	√	√	√	
	PFCE6	R/W	0000_H	16	—	—	—	—	√	√	√	√	√	√	√	√	√	√	√	√	
	PFCAE6	R/W	0000_H	16	—	—	—	—	√	√	√	√	√	√	√	√	√	√	√	√	
	PNOT6	W	0000_H	16	—	—	—	—	√	√	√	√	√	√	√	√	√	√	√	√	
	PMSR6	R/W	$0000\ FFFF_H$	32	—	—	—	—	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
					—	—	—	—	√	√	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
	PMCSR6	R/W	$0000\ 0000_H$	32	—	—	—	—	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
					—	—	—	—	√	√	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
	PIBC6	R/W	0000_H	16	—	—	—	—	√	√	√	√	√	√	√	√	√	√	√	√	
	PBDC6	R/W	0000_H	16	—	—	—	—	√	√	√	√	√	√	√	√	√	√	√	√	
	PIPC6	R/W	0000_H	16	—	—	—	—	√	√	√	√	√	√	√	√	√	√	—	—	
	PU6	R/W	0000_H	16	—	—	—	—	√	√	√	√	√	√	√	√	√	√	√	√	
	PD6	R/W	0000_H	16	—	—	—	—	√	√	√	√	√	√	√	√	√	√	√	√	
	PDSC6	R/W	$0000\ 0000_H$	32	—	—	—	—	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
					—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	Upper 16 bits
	PUCC6	R/W	$0000\ 0000_H$	32	—	—	—	—	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
					—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	Upper 16 bits
	PPROTS6	R	00_H	8	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	√	
	PPCMD6	W	00_H	8	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
	PINV6	R/W	$0000\ 0000_H$	32	—	—	—	—	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
					—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	Upper 16 bits

Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap 30 PINV	25 PUCC	24 PDSC	19 PU	18 PD	17 PBDC	16 PIBC	12 P	8 PPR	6 PMC	5 PIPC	4 PM	2 PFCAE	1 PFCE	0 PFC
6	PCR6_0	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	—	√	√	√	√
	PCR6_1	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	—	√	√	√	√
	PCR6_2	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR6_3	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR6_4	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR6_5	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR6_6	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR6_7	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR6_8	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR6_9	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR6_10	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR6_11	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√

Table 2.37 List of Registers in E1M-S Port Group 7

Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Remarks
7	P7	R/W	0000_H	16	—	—	—	—	√	√	√	√	√	√	√	√	√	√	√	√	
	PSR7	R/W	$0000\ 0000_H$	32	—	—	—	—	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
					—	—	—	—	√	√	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
	PPR7	R	0000_H	16	—	—	—	—	√	√	√	√	√	√	√	√	√	√	√	√	
	PM7	R/W	$FFFF_H$	16	—	—	—	—	√	√	√	√	√	√	√	√	√	√	√	√	
	PMC7	R/W	0000_H	16	—	—	—	—	√	√	√	√	√	√	√	√	√	√	√	√	
	PFC7	R/W	0000_H	16	—	—	—	—	√	√	√	√	√	√	√	√	√	√	√	√	
	PFCE7	R/W	0000_H	16	—	—	—	—	√	√	√	√	√	√	√	√	√	√	√	√	
	PFCAE7	R/W	0000_H	16	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	
	PNOT7	W	0000_H	16	—	—	—	—	√	√	√	√	√	√	√	√	√	√	√	√	
	PMSR7	R/W	$0000\ FFFF_H$	32	—	—	—	—	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
					—	—	—	—	√	√	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
	PMCSR7	R/W	$0000\ 0000_H$	32	—	—	—	—	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
					—	—	—	—	√	√	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
	PIBC7	R/W	0000_H	16	—	—	—	—	√	√	√	√	√	√	√	√	√	√	√	√	
	PBDC7	R/W	0000_H	16	—	—	—	—	√	√	√	√	√	√	√	√	√	√	√	√	
	PIPC7	R/W	0000_H	16	—	—	—	—	√	√	√	√	√	√	√	√	√	√	√	√	
	PU7	R/W	0000_H	16	—	—	—	—	√	√	√	√	√	√	√	√	√	√	√	√	
	PD7	R/W	0000_H	16	—	—	—	—	√	√	√	√	√	√	√	√	√	√	√	√	
	PDSC7	R/W	$0000\ 0000_H$	32	—	—	—	—	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
					—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	Upper 16 bits
	PUCC7	R/W	$0000\ 0000_H$	32	—	—	—	—	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
					—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	Upper 16 bits
	PPROTS7	R	00_H	8	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	√	
	PPCMD7	W	00_H	8	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
	PINV7	R/W	$0000\ 0000_H$	32	—	—	—	—	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
					—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	Upper 16 bits

Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap 30 PINV	25 PUCC	24 PDSC	19 PU	18 PD	17 PBDC	16 PIBC	12 P	8 PPR	6 PMC	5 PIPC	4 PM	2 PFCAE	1 PFCE	0 PFC
7	PCR7_0	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR7_1	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR7_2	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR7_3	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR7_4	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR7_5	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR7_6	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR7_7	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR7_8	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR7_9	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	—	√	√
	PCR7_10	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	—	√	√
	PCR7_11	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	—	√	√

Table 2.38 List of Registers in E1M-S Port Group 8

Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Remarks
8	P8	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
	PSR8	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Lower 16 bits
					—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Upper 16 bits
	PPR8	R	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
	PM8	R/W	$FFFF_H$	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
	PMC8	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
	PFC8	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
	PFCE8	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
	PFCAE8	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	—	—	√	√	√	√	√	
	PNOT8	W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
	PMSR8	R/W	$0000\ FFFF_H$	32	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Lower 16 bits
					—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Upper 16 bits
	PMCSR8	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Lower 16 bits
					—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Upper 16 bits
	PIBC8	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
	PBDC8	R/W	0000H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
	PIPC8	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
	PU8	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
	PD8	R/W	0000_H	16	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
	PDSC8	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Lower 16 bits
					—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	Upper 16 bits
	PUCC8	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Lower 16 bits
					—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	Upper 16 bits
	PPROTS8	R	00_H	8	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	√	
	PPCMD8	W	00_H	8	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
	PINV8	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	Lower 16 bits
					—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	Upper 16 bits

Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap 30 PINV	25 PUCC	24 PDSC	19 PU	18 PD	17 PBDC	16 PIBC	12 P	8 PPR	6 PMC	5 PIPC	4 PM	2 PFCAE	1 PFCE	0 PFC
8	PCR8_0	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR8_1	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR8_2	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR8_3	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR8_4	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR8_5	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	—	√	√
	PCR8_6	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	—	√	√
	PCR8_7	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√

Table 2.39 List of Registers in E1M-S Port Group 10

Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Remarks
10	P10	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	
	PSR10	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	Lower 16 bits
					—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	Upper 16 bits
	PPR10	R	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	
	PM10	R/W	$FFFF_H$	16	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	
	PMC10	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	
	PFC10	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	√	—	—	√	√	
	PFCE10	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	—	
	PFCAE10	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	
	PNOT10	W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	
	PMSR10	R/W	$0000\ FFFF_H$	32	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	Lower 16 bits
					—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	Upper 16 bits
	PMCSR10	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	Lower 16 bits
					—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	Upper 16 bits
	PIBC10	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	
	PBDC10	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	
	PIPC10	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	
	PU10	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	
	PD10	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	
	PDSC10	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	Lower 16 bits
					—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	Upper 16 bits
	PUCC10	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	Lower 16 bits
					—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	Upper 16 bits
	PPROTS10	R	00_H	8	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	√	
	PPCMD10	W	00_H	8	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
	PINV10	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	Lower 16 bits
					—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	Upper 16 bits

Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap 30 PINV	25 PUCC	24 PDSC	19 PU	18 PD	17 PBDC	16 PIBC	12 P	8 PPR	6 PMC	5 PIPC	4 PM	2 PFCAE	1 PFCE	0 PFC
10	PCR10_0	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	—	√
	PCR10_1	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
	PCR10_2	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	—
	PCR10_3	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	—
	PCR10_4	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√

Table 2.40 List of Registers in E1M-S Port Group 11

Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Remarks
11	P11	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	√	
	PSR11	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	√	Lower 16 bits
					—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	√	Upper 16 bits
	PPR11	R	0000_H	16	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	√	
	PM11	R/W	$FFFF_H$	16	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	√	
	PMC11	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	—	
	PFC11	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	—	
	PFCE11	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	—	
	PNOT11	W	0000_H	16	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	√	
	PMSR11	R/W	$0000\ FFFF_H$	32	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	√	Lower 16 bits
					—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	√	Upper 16 bits
	PMCSR11	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	—	Lower 16 bits
					—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	—	Upper 16 bits
	PIBC11	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	√	
	PBDC11	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	√	
	PU11	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	√	
	PD11	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	√	
	PDSC11	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	√	Lower 16 bits
					—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	Upper 16 bits
	PUCC11	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	√	Lower 16 bits
					—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	Upper 16 bits
	PPROTS11	R	00_H	8	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	√	
	PPCMD11	W	00_H	8	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
	PINV11	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	√	Lower 16 bits
					—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	Upper 16 bits

Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap 30 PINV	25 PUCC	24 PDSC	19 PU	18 PD	17 PBDC	16 PIBC	12 P	8 PPR	6 PMC	5 PIPC	4 PM	2 PFCAE	1 PFCE	0 PFC
11	PCR11_0	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	—	—	√	—	—	—
	PCR11_1	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	—	√	—	√	√
	PCR11_2	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	—	√	—	√	√
	PCR11_3	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	—	√	—	√	√
	PCR11_4	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	—	√	—	√	√
	PCR11_5	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	—	√	—	√	√

Table 2.41 List of Registers in E1M-S Port Group 13

Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Remarks
13	P13	R/W	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
	PSR13	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
					—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
	PPR13	R	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
	PM13	R/W	$FFFF_H$	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
	PMC13	R/W	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
	PFC13	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	√	
	PFCE13	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	√	—	—	—	—	
	PFCAE13	R/W	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
	PNOT13	W	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
	PMSR13	R/W	$0000\ FFFF_H$	32	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
					—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
	PMCSR13	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
					—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
	PIBC13	R/W	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
	PBDC13	R/W	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
	PIPC13	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	—	—	√	√	—	
	PU13	R/W	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
	PD13	R/W	0000_H	16	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	
	PDSC13	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
					—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	Upper 16 bits
	PUCC13	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
					—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	Upper 16 bits
	PPROTS13	R	00_H	8	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	√	
	PPCMD13	W	00_H	8	—	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	
	PINV13	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
					—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	Upper 16 bits

Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap 30 PINV	25 PUCC	24 PDSC	19 PU	18 PD	17 PBDC	16 PIBC	12 P	8 PPR	6 PMC	5 PIPC	4 PM	2 PFCAE	1 PFCE	0 PFC
13	PCR13_0	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	—	√	√	—	√
	PCR13_1	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	—	√
	PCR13_2	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	—	√
	PCR13_3	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	—	√	√	—	√
	PCR13_4	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	—	√	√	√	√
	PCR13_5	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	—	√	√	—	—
	PCR13_6	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	—	√	√	—	—
	PCR13_7	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	—	√	√	—	—
	PCR13_8	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	—	√	√	—	—
	PCR13_9	R/W	$0000\ 0010_H$	32	√	√	√	√	√	√	√	√	√	√	—	√	√	—	—

Table 2.42 List of Registers in E1M-S Port Group 14

Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Remarks
14	PPR14	R	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
	PMC14	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	
	PFC14	R/W	0000_H	16	√	—	√	—	√	√	√	√	√	—	√	√	√	—	—	—	
	PMCSR14	R/W	$0000\ 0000_H$	32	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Lower 16 bits
					√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	Upper 16 bits
	PIBC14	R/W	0000_H	16	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	

Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap 30 PINV	25 PUCC	24 PDSC	19 PU	18 PD	17 PBDC	16 PIBC	12 P	8 PPR	6 PMC	5 PIPC	4 PM	2 PFCAE	1 PFCE	0 PFC
14	PCR14_0	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	√	—	√	√	—	—	—	—	—
	PCR14_1	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	√	—	√	√	—	—	—	—	—
	PCR14_2	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	√	—	√	√	—	—	—	—	—
	PCR14_3	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	√	—	√	√	—	—	—	—	√
	PCR14_4	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	√	—	√	√	—	—	—	—	√
	PCR14_5	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	√	—	√	√	—	—	—	—	√
	PCR14_6	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	√	—	√	√	—	—	—	—	—
	PCR14_7	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	√	—	√	√	—	—	—	—	√
	PCR14_8	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	√	—	√	√	—	—	—	—	√
	PCR14_9	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	√	—	√	√	—	—	—	—	√
	PCR14_10	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	√	—	√	√	—	—	—	—	√
	PCR14_11	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	√	—	√	√	—	—	—	—	√
	PCR14_12	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	√	—	√	√	—	—	—	—	—
	PCR14_13	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	√	—	√	√	—	—	—	—	√
	PCR14_14	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	√	—	√	√	—	—	—	—	—
	PCR14_15	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	√	—	√	√	—	—	—	—	√

Table 2.43 List of Registers in E1M-S Port Group 15

Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Remarks
15	PPR15	R	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	
	PMC15	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	
	PFC15	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	√	
	PMCSR15	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	Lower 16 bits
					—	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	Upper 16 bits
	PIBC15	R/W	0000_H	16	—	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	

Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap 30 PINV	25 PUCC	24 PDSC	19 PU	18 PD	17 PBDC	16 PIBC	12 P	8 PPR	6 PMC	5 PIPC	4 PM	2 PFCAE	1 PFCE	0 PFC
15	PCR15_0	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	√	—	√	√	—	—	—	—	√
	PCR15_1	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	√	—	√	√	—	—	—	—	—
	PCR15_2	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	√	—	√	√	—	—	—	—	—
	PCR15_3	R/W	$0000\ 0000_H$	32	—	—	—	—	—	—	√	—	√	√	—	—	—	—	—

Table 2.44 List of Registers in E1M-S Port Group 16

Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Remarks
16	PPR16	R	0000_H	16	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	
	PIBC16	R/W	0000_H	16	—	—	—	—	—	—	—	√	√	√	√	√	√	√	√	√	

Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap 30 PINV	25 PUCC	24 PDSC	19 PU	18 PD	17 PBDC	16 PIBC	12 P	8 PPR	6 PMC	5 PIPC	4 PM	2 PFCAE	1 PFCE	0 PFC
16	PCR16_0	R/W	0000 0000_H	32	—	—	—	—	—	—	√	—	√	—	—	—	—	—	—
	PCR16_1	R/W	0000 0000_H	32	—	—	—	—	—	—	√	—	√	—	—	—	—	—	—
	PCR16_2	R/W	0000 0000_H	32	—	—	—	—	—	—	√	—	√	—	—	—	—	—	—
	PCR16_3	R/W	0000 0000_H	32	—	—	—	—	—	—	√	—	√	—	—	—	—	—	—
	PCR16_4	R/W	0000 0000_H	32	—	—	—	—	—	—	√	—	√	—	—	—	—	—	—
	PCR16_5	R/W	0000 0000_H	32	—	—	—	—	—	—	√	—	√	—	—	—	—	—	—
	PCR16_6	R/W	0000 0000_H	32	—	—	—	—	—	—	√	—	√	—	—	—	—	—	—
	PCR16_7	R/W	0000 0000_H	32	—	—	—	—	—	—	√	—	√	—	—	—	—	—	—
	PCR16_8	R/W	0000 0000_H	32	—	—	—	—	—	—	√	—	√	—	—	—	—	—	—

Table 2.45 List of Registers in E1M-S Port Group 17

Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap 15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Remarks
17	PM17	R/W	$FFFF_H$	16	—	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	
	PMSR17	R/W	0000 $FFFF_H$	32	—	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	Lower 16 bits
					—	—	—	—	—	—	—	—	—	—	—	—	√	√	√	√	Upper 16 bits

Port Group Name	Register Name	R/W	Value after Reset	Access Size	Bitmap 30 PINV	25 PUCC	24 PDSC	19 PU	18 PD	17 PBDC	16 PIBC	12 P	8 PPR	6 PMC	5 PIPC	4 PM	2 PFCAE	1 PFCE	0 PFC
17	PCR17_0	R/W	0000 0010_H	32	—	—	—	—	—	—	—	—	—	—	—	√	—	—	—
	PCR17_1	R/W	0000 0010_H	32	—	—	—	—	—	—	—	—	—	—	—	√	—	—	—
	PCR17_2	R/W	0000 0010_H	32	—	—	—	—	—	—	—	—	—	—	—	√	—	—	—
	PCR17_3	R/W	0000 0010_H	32	—	—	—	—	—	—	—	—	—	—	—	√	—	—	—

2.2.1.2 Alternative Function List of the E1M-S Pins

Table 2.46 to **Table 2.61** show the list of alternative functions of each port pin. In the tables, " — " means a reserved bit which cannot be selected.

The pins of port groups 7, 8, and 10 is included in the BGA304 package, but not in BGA252. The setting of port registers should not be modified (the settings should be fixed to their values after reset) because the functions of the port groups, which are not included, cannot be used.

Table 2.46 List of Pin Alternative Functions in E1M-S Port Group 0

Port	General Input/ Output Port	1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative		Dedicated Function
		ALT_IN1	ALT_OUT1	ALT_IN2	ALT_OUT2	ALT_IN3	ALT_OUT3	ALT_IN4	ALT_OUT4	ALT_IN5	ALT_OUT5	ALT_IN6	ALT_OUT6	ALT_IN7	ALT_OUT7	
P0_0	IO	—	—	—	—	—	—	FR0RXDA	—	CSIH0TSI	—	SCI0RxD	—	—	—	—
P0_1	IO	—	—	—	—	—	—	—	FR0TXDA	—	CSIH0TSO	—	SCI0TxD	—	—	—
P0_2	IO	—	—	—	—	—	—	—	$\overline{\text{FR0ENA}}$	CSIH0TSCK	CSIH0TSCK	SCI0SCK	SCI0SCK	—	—	—
P0_3	IO	—	—	—	—	—	—	FR0RXDB	—	—	CSIH0TCSS0	CRX2	—	IRQ0	—	—
P0_4	IO	—	—	—	—	—	—	—	FR0TXDB	—	CSIH0TCSS1	—	CTX2	IRQ1	—	—
P0_5	IO	—	—	—	—	—	—	—	$\overline{\text{FR0ENB}}$	—	CSIH0TCSS2	—	—	IRQ2	—	—
P0_6	IO	—	—	—	—	—	—	—	—	—	—	CSIH1TSI	—	IRQ3	—	—
P0_7	IO	—	—	—	—	—	—	—	—	—	—	—	CSIH1TSO	IRQ4	—	—
P0_8	IO	TIA00	—	—	—	—	—	—	—	—	—	CSIH1TSCK	CSIH1TSCK	IRQ5	—	—
P0_9	IO	TIA01	—	—	—	—	—	—	—	$\overline{\text{CSIH0TSSI}}$	—	—	CSIH1TCSS0	IRQ6	—	—
P0_10	IO	TIA02	—	—	—	—	—	—	—	CSIH0TRY	CSIH0TRY	—	CSIH1TCSS1	IRQ7	—	—
P0_11	IO	TIA03	—	—	TOD00B	—	—	—	—	—	CSIH0TCSS3	—	CSIH1TCSS2	CSIH2TSI	—	—
P0_12	IO	TIA04	—	—	TOD01B	—	—	—	—	—	CSIH0TCSS4	—	CSIH1TCSS3	—	CSIH2TSO	—
P0_13	IO	TIA05	—	—	TOD02B	—	—	—	—	—	CSIH0TCSS5	$\overline{\text{CSIH1TSSI}}$	—	CSIH2TSCK	CSIH2TSCK	—
P0_14	IO	TIA06	—	—	TOD03B	—	—	—	—	—	—	CSIH1TRY	CSIH1TRY	—	CSIH2TCSS0	—

Table 2.47 List of Pin Alternative Functions in E1M-S Port Group 1 (1/2)

Port	General Input/ Output Port	1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative		Dedicated Function
		ALT_IN1	ALT_OUT1	ALT_IN2	ALT_OUT2	ALT_IN3	ALT_OUT3	ALT_IN4	ALT_OUT4	ALT_IN5	ALT_OUT5	ALT_IN6	ALT_OUT6	ALT_IN7	ALT_OUT7	
P1_0	IO	—	TOD20A	—	TOD20B	—	TOE00	—	APA0	—	—	SCI0RxD	—	—	CSIH2TCSS1	—
P1_1	IO	—	TOD21A	—	TOD21B	—	TOE01	—	APA1	—	TSO00	—	SCI0TxD	—	CSIH2TCSS2	—
P1_2	IO	—	TOD22A	—	TOD22B	—	TOE02	—	APA2	—	TSO01	—	—	—	CSIH2TCSS3	—
P1_3	IO	—	TOD23A	—	TOD23B	—	TOE03	—	APA3	—	TSO02	—	—	$\overline{\text{CSIH2TSSI}}$	—	—
P1_4	IO	—	TOD30A	—	TOD30B	—	TOE40	—	APA4	—	TSO03	—	—	CSIH2TRY	CSIH2TRY	—

Table 2.47 List of Pin Alternative Functions in E1M-S Port Group 1 (2/2)

Port	General Input/ Output Port	1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative		Dedicated Function
		ALT_IN1	ALT_OUT1	ALT_IN2	ALT_OUT2	ALT_IN3	ALT_OUT3	ALT_IN4	ALT_OUT4	ALT_IN5	ALT_OUT5	ALT_IN6	ALT_OUT6	ALT_IN7	ALT_OUT7	
P1_5	IO	—	TOD31A	—	TOD31B	—	TOE41	—	APA5	—	TSO04	—	—	—	—	—
P1_6	IO	—	TOD32A	—	TOD32B	—	TOE42	—	APA6	—	TSO05	—	—	—	—	—
P1_7	IO	—	TOD33A	—	TOD33B	—	TOE43	—	APA7	—	TSO06	—	—	—	—	—
P1_8	IO	—	TOD50A	—	TOD50B	—	TOE20	—	APA8	—	TSO07	—	—	LRX0	—	—
P1_9	IO	—	TOD51A	—	TOD51B	—	TOE21	—	APA9	—	TSO10	—	—	—	LTX0	—
P1_10	IO	—	TOD52A	—	TOD52B	—	TOE22	—	APA10	—	TSO11	—	—	—	—	—
P1_11	IO	—	TOD53A	—	TOD53B	—	TOE23	—	APA11	—	TSO12	—	—	—	—	—
P1_12	IO	TIOC50	TIOC50	—	TOD60B	—	TOE30	—	APA12	—	TSO13	—	—	—	—	—
P1_13	IO	TIOC51	TIOC51	—	TOD61B	—	TOE31	—	APA13	—	TSO14	—	—	—	—	—
P1_14	IO	TIOC52	TIOC52	—	TOD62B	—	TOE32	—	APA14	—	TSO15	—	—	—	—	—
P1_15	IO	TIOC53	TIOC53	—	TOD63B	—	TOE33	—	APA15	—	TSO16	—	—	—	—	—

Table 2.48 List of Pin Alternative Functions in E1M-S Port Group 2

Port	General Input/ Output Port	1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative		Dedicated Function
		ALT_IN1	ALT_OUT1	ALT_IN2	ALT_OUT2	ALT_IN3	ALT_OUT3	ALT_IN4	ALT_OUT4	ALT_IN5	ALT_OUT5	ALT_IN6	ALT_OUT6	ALT_IN7	ALT_OUT7	
P2_0	IO	TIA00	TOE50	—	TOD00B	TIOC12	TIOC12	—	—	—	TSO17	—	—	IRQ0	—	—
P2_1	IO	TIA01	TOE51	—	TOD01B	TIOC13	TIOC13	—	—	—	—	—	—	IRQ1	—	—
P2_2	IO	TIA02	TOE52	—	TOD02B	TIOC20	TIOC20	—	—	—	—	—	—	IRQ2	—	—
P2_3	IO	TIA03	TOE53	—	TOD03B	TIOC21	TIOC21	—	—	—	—	—	—	IRQ3	—	—
P2_4	IO	TIA04	TOE10	—	TOD10B	TIOC22	TIOC22	—	—	CRX3	—	—	—	IRQ4	—	—
P2_5	IO	TIA05	TOE11	—	TOD11B	TIOC23	TIOC23	—	—	—	CTX3	—	—	IRQ5	—	—
P2_6	IO	TCLKA	TOE12	—	TOD12B	TIOC30	TIOC30	—	—	CRX0	—	—	—	IRQ6	—	—
P2_7	IO	TCLKB	TOE13	—	TOD13B	TIOC31	TIOC31	—	—	—	CTX0	—	—	IRQ7	—	—

Table 2.49 List of Pin Alternative Functions in E1M-S Port Group 3 (1/2)

Port	General Input/ Output Port	1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative		Dedicated Function
		ALT_IN1	ALT_OUT1	ALT_IN2	ALT_OUT2	ALT_IN3	ALT_OUT3	ALT_IN4	ALT_OUT4	ALT_IN5	ALT_OUT5	ALT_IN6	ALT_OUT6	ALT_IN7	ALT_OUT7	
P3_0	IO	TIF0A	TOE60	—	TOD30B	TIOC32	TIOC32	—	—	CRX1	—	SCI1RxD	—	—	—	—
P3_1	IO	TIF1A	TOE61	—	TOD31B	TIOC33	TIOC33	—	—	—	CTX1	—	SCI1TxD	—	—	—
P3_2	IO	TIF2A	TOE62	—	TOD32B	TIOC40	TIOC40	ADTRG1	—	CRX2	—	SCI2RxD	—	—	—	—
P3_3	IO	TIF3	TOE63	—	TOD33B	TIOC41	TIOC41	—	ADEND1	—	CTX2	ESO2	SCI2TxD	—	—	—
P3_4	IO	TIF4	—	TIJ2	TOD50B	TIOC42	TIOC42	ADTRG0	—	—	—	ESO3	—	—	—	—

Table 2.49 List of Pin Alternative Functions in E1M-S Port Group 3 (2/2)

Port	General Input/ Output Port	1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative		Dedicated Function
		ALT_IN1	ALT_OUT1	ALT_IN2	ALT_OUT2	ALT_IN3	ALT_OUT3	ALT_IN4	ALT_OUT4	ALT_IN5	ALT_OUT5	ALT_IN6	ALT_OUT6	ALT_IN7	ALT_OUT7	
P3_5	IO	TIF5	—	TIJ3	TOD51B	TIOC43	TIOC43	—	ADEND0	—	—	—	—	—	—	—
P3_6	IO	TIA06	—	—	TOD52B	TIOC00	TIOC00	—	—	CRX3	—	—	—	—	—	—
P3_7	IO	TIF6	—	—	TOD53B	TIOC01	TIOC01	—	—	—	CTX3	—	—	—	—	—

Table 2.50 List of Pin Alternative Functions in E1M-S Port Group 4

Port	General Input/ Output Port	1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative		Dedicated Function
		ALT_IN1	ALT_OUT1	ALT_IN2	ALT_OUT2	ALT_IN3	ALT_OUT3	ALT_IN4	ALT_OUT4	ALT_IN5	ALT_OUT5	ALT_IN6	ALT_OUT6	ALT_IN7	ALT_OUT7	
P4_0	IO	—	TOD00A	TAPTS00	TOD00B	TIOC20	TIOC20	TIF0A	APA0	—	$\overline{\text{ERROROUT\_C}}$	—	—	CSIH3TSI	—	—
P4_1	IO	—	TOD01A	TAPTS01	TOD01B	TIOC21	TIOC21	TIF1A	APA1	—	—	—	—	—	CSIH3TSO	—
P4_2	IO	—	TOD02A	TAPTS02	TOD02B	TIOC22	TIOC22	TIF2A	APA2	—	—	—	—	CSIH3TSCK	CSIH3TSCK	—
P4_3	IO	—	TOD03A	TAPTS10	TOD03B	TIOC23	TIOC23	TIF3	APA3	—	—	$\overline{\text{CSIH2TSSI}}$	—	—	CSIH3TCSS0	—
P4_4	IO	—	TOD10A	TAPTS11	TOD10B	—	TOE30	TIF4	APA4	—	—	CSIH2TRY	CSIH2TRY	—	CSIH3TCSS1	—
P4_5	IO	—	TOD11A	TAPTS12	TOD11B	—	TOE31	TIF5	APA5	CRX0	—	—	—	—	CSIH3TCSS2	—
P4_6	IO	—	TOD12A	—	TOD12B	—	TOE32	TIF6	APA6	—	CTX0	—	—	—	CSIH3TCSS3	—
P4_7	IO	—	TOD13A	—	TOD13B	—	TOE00	TIF7	APA7	—	—	CSIH2TSI	—	$\overline{\text{CSIH3TSSI}}$	—	—
P4_8	IO	—	TOD40A	—	TOD40B	—	TOE01	TIF8	APA8	—	—	—	CSIH2TSO	CSIH3TRY	CSIH3TRY	—
P4_9	IO	—	TOD41A	—	TOD41B	—	TOE02	TIF9	APA9	CRX3	—	CSIH2TSCK	CSIH2TSCK	—	—	—
P4_10	IO	—	TOD42A	—	TOD42B	—	TOE03	TIF10	APA10	—	CTX3	—	CSIH2TCSS0	—	—	—
P4_11	IO	—	TOD43A	—	TOD43B	—	TOE10	TIF11	APA11	—	—	—	CSIH2TCSS1	—	—	—
P4_12	IO	TIOC42	TIOC42	—	TOD50B	—	TOE11	TIF12	APA12	SCI1RxD	—	—	CSIH2TCSS2	—	—	—
P4_13	IO	TIOC43	TIOC43	—	TOD51B	—	TOE12	TIF13	APA13	—	SCI1TxD	—	CSIH2TCSS3	—	—	—
P4_14	IO	—	TOD20B	—	TOD52B	—	TOE13	TIF14	APA14	CRX1	—	—	—	—	—	—
P4_15	IO	—	TOD21B	—	TOD53B	—	TOE40	TIF15	APA15	—	CTX1	—	—	—	$\overline{\text{ERROROUT\_C}}$	—

Table 2.51 List of Pin Alternative Functions in E1M-S Port Group 5

Port	General Input/ Output Port	1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative		Dedicated Function
		ALT_IN1	ALT_OUT1	ALT_IN2	ALT_OUT2	ALT_IN3	ALT_OUT3	ALT_IN4	ALT_OUT4	ALT_IN5	ALT_OUT5	ALT_IN6	ALT_OUT6	ALT_IN7	ALT_OUT7	
P5_0	IO	TIF0B	TOE41	—	TOD20B	TIOC00	TIOC00	—	—	CRX0	—	SCI1RxD	—	—	—	—
P5_1	IO	TIF1B	TOE42	—	TOD21B	TIOC01	TIOC01	—	—	—	CTX0	—	SCI1TxD	—	—	—
P5_2	IO	TIF2B	TOE43	—	TOD22B	TIOC02	TIOC02	ESO2	—	SCI0RxD	—	SCI1SCK	SCI1SCK	$\overline{\text{POD}}$	—	—
P5_3	IO	TIF6	—	TIJ0	TOD23B	TIOC03	TIOC03	ESO3	—	CRX1	—	SCI2RxD	—	—	—	—
P5_4	IO	TIF7	—	TIJ1	TOD30B	TIOC10	TIOC10	—	—	—	CTX1	—	SCI2TxD	DSADTRG 1	—	—
P5_5	IO	TIF8	TOE20	—	TOD31B	TIOC11	TIOC11	—	—	—	SCI0TxD	SCI2SCK	SCI2SCK	—	DSADEND 1	—
P5_6	IO	TIF9	TOE21	—	TOD32B	TIOC12	TIOC12	LRX0	—	CRX2	—	—	—	DSADTRG 0	—	—
P5_7	IO	TIF10	TOE22	—	TOD33B	TIOC13	TIOC13	—	LTX0	—	CTX2	SCI3SCK	SCI3SCK	—	DSADEND 0	—
P5_8	IO	TIF11	TOE23	TIJ2	—	TIOC40	TIOC40	—	—	—	—	SCI3RxD	—	DSADTRG 2	—	—
P5_9	IO	TIF12	TOE33	TIJ3	—	TIOC41	TIOC41	—	—	—	—	—	SCI3TxD	—	DSADEND 2	—

Table 2.52 List of Pin Alternative Functions in E1M-S Port Group 6 (1/2)

Port	General Input/ Output Port	1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative		Dedicated Function
		ALT_IN1	ALT_OUT1	ALT_IN2	ALT_OUT2	ALT_IN3	ALT_OUT3	ALT_IN4	ALT_OUT4	ALT_IN5	ALT_OUT5	ALT_IN6	ALT_OUT6	ALT_IN7	ALT_OUT7	
P6_0	IO	TIF13	TOD60A	TIJ0	—	—	—	—	TOE40	CRX0	—	—	—	DSADTRG 3	—	—
P6_1	IO	TIF14	TOD61A	TIJ1	—	—	—	—	TOE41	—	CTX0	—	—	—	DSADEND 3	—
P6_2	IO	TIF15	TOD62A	TIJ2	—	—	—	—	TOE42	CRX3	—	SCI1RxD	—	DSADTRG 4	—	—
P6_3	IO	TIF0A	TOD63A	—	—	TIOC60	TIOC60	—	TOE43	—	CTX3	—	SCI1TxD	—	DSADEND 4	—
P6_4	IO	—	TOD70A	—	TOD60B	TIOC61	TIOC61	—	TOE50	CRX1	—	—	—	DSADTRG 5	—	—
P6_5	IO	—	TOD71A	—	TOD61B	TIOC62	TIOC62	—	TOE51	—	CTX1	—	—	—	DSADEND 5	—
P6_6	IO	—	TOD72A	—	TOD62B	TIOC63	TIOC63	—	TOE52	—	—	SCI2RxD	—	DSADTRG 6	—	—
P6_7	IO	—	TOD73A	TIJ3	TOD63B	—	—	—	TOE53	—	—	—	SCI2TxD	—	DSADEND 6	—
P6_8	IO	TIF1A	—	—	TOD70B	TIOC70	TIOC70	—	TOE60	—	—	SCI3RxD	—	DSADTRG 7	—	—

Table 2.52 List of Pin Alternative Functions in E1M-S Port Group 6 (2/2)

Port	General Input/ Output Port	1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative		Dedicated Function
		ALT_IN1	ALT_OUT1	ALT_IN2	ALT_OUT2	ALT_IN3	ALT_OUT3	ALT_IN4	ALT_OUT4	ALT_IN5	ALT_OUT5	ALT_IN6	ALT_OUT6	ALT_IN7	ALT_OUT7	
P6_9	IO	TIF2A	—	—	TOD71B	TIOC71	TIOC71	—	TOE61	—	—	—	SCI3TxD	—	DSADEND7	—
P6_10	IO	—	—	TIJ4	TOD72B	TIOC72	TIOC72	—	TOE62	—	—	SCI0RxD	—	—	—	—
P6_11	IO	—	—	TIJ5	TOD73B	TIOC73	TIOC73	—	TOE63	—	—	—	SCI0TxD	—	—	—

Table 2.53 List of Pin Alternative Functions in E1M-S Port Group 7

Port	General Input/ Output Port	1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative		Dedicated Function
		ALT_IN1	ALT_OUT1	ALT_IN2	ALT_OUT2	ALT_IN3	ALT_OUT3	ALT_IN4	ALT_OUT4	ALT_IN5	ALT_OUT5	ALT_IN6	ALT_OUT6	ALT_IN7	ALT_OUT7	
P7_0	IO	—	—	—	—	TIOC00	TIOC00	—	—	—	—	CSIH3TSI	—	—	—	—
P7_1	IO	—	—	—	—	TIOC01	TIOC01	—	—	—	—	—	CSIH3TSO	—	—	—
P7_2	IO	—	—	—	—	TIOC02	TIOC02	—	—	—	—	CSIH3TSCK	CSIH3TSCK	—	—	—
P7_3	IO	—	—	—	—	TIOC03	TIOC03	—	—	—	—	—	CSIH3TCSS0	—	—	—
P7_4	IO	—	TOD60A	—	TOD60B	TIOC10	TIOC10	—	—	—	—	—	CSIH3TCSS1	—	—	—
P7_5	IO	—	TOD61A	TIJ0	TOD61B	TIOC11	TIOC11	—	—	—	—	—	CSIH3TCSS2	—	—	—
P7_6	IO	—	TOD62A	TIJ1	TOD62B	TIOC12	TIOC12	—	—	—	—	—	CSIH3TCSS3	—	—	—
P7_7	IO	—	TOD63A	—	TOD63B	TIOC13	TIOC13	—	TOE00	—	—	$\overline{\text{CSIH3TSSI}}$	—	—	—	—
P7_8	IO	—	TOD70A	—	TOD70B	TIOC20	TIOC20	—	TOE01	—	—	CSIH3TRY	CSIH3TRY	—	—	—
P7_9	IO	—	TOD71A	—	TOD71B	TIOC21	TIOC21	—	TOE02	—	—	—	—	—	—	—
P7_10	IO	—	TOD72A	—	TOD72B	TIOC22	TIOC22	—	TOE03	—	—	—	—	—	—	—
P7_11	IO	—	TOD73A	—	TOD73B	TIOC23	TIOC23	—	—	—	—	—	—	—	—	—

Table 2.54 List of Pin Alternative Functions in E1M-S Port Group 8 (1/2)

Port	General Input/ Output Port	1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative		Dedicated Function
		ALT_IN1	ALT_OUT1	ALT_IN2	ALT_OUT2	ALT_IN3	ALT_OUT3	ALT_IN4	ALT_OUT4	ALT_IN5	ALT_OUT5	ALT_IN6	ALT_OUT6	ALT_IN7	ALT_OUT7	
P8_0	IO	TIF5	TOE10	—	—	TIOC60	TIOC60	—	—	—	CTX2	—	SCI1TxD	—	—	—
P8_1	IO	TIF6	TOE11	—	—	TIOC61	TIOC61	—	—	—	—	SCI2RxD	—	—	—	—
P8_2	IO	TIF7	TOE12	—	—	TIOC62	TIOC62	—	—	—	—	—	SCI2TxD	—	—	—
P8_3	IO	TIF8	TOE13	—	—	TIOC63	TIOC63	—	—	—	—	SCI0RxD	—	—	—	—
P8_4	IO	TIF9	TOE20	TIJ0	—	TIOC70	TIOC70	—	—	—	—	—	SCI0TxD	—	—	—

Table 2.54 List of Pin Alternative Functions in E1M-S Port Group 8 (2/2)

Port	General Input/ Output Port	1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative		Dedicated Function
		ALT_IN1	ALT_OUT1	ALT_IN2	ALT_OUT2	ALT_IN3	ALT_OUT3	ALT_IN4	ALT_OUT4	ALT_IN5	ALT_OUT5	ALT_IN6	ALT_OUT6	ALT_IN7	ALT_OUT7	
P8_5	IO	TIF10	TOE21	TIJ1	—	TIOC71	TIOC71	—	—	—	—	—	—	—	—	—
P8_6	IO	TIF11	TOE22	TIJ2	—	TIOC72	TIOC72	—	—	—	—	—	—	—	—	—
P8_7	IO	TIF12	TOE23	TIJ3	—	TIOC73	TIOC73	—	—	—	—	—	—	IRQ0	—	—

Table 2.55 List of Pin Alternative Functions in E1M-S Port Group 10

Port	General Input/ Output Port	1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative		Dedicated Function
		ALT_IN1	ALT_OUT1	ALT_IN2	ALT_OUT2	ALT_IN3	ALT_OUT3	ALT_IN4	ALT_OUT4	ALT_IN5	ALT_OUT5	ALT_IN6	ALT_OUT6	ALT_IN7	ALT_OUT7	
P10_0	IO	TIF0A	—	—	—	—	—	—	—	CRX0	—	SCI3RxD	—	—	—	—
P10_1	IO	TIF1A	TOE00	—	—	TIOC50	TIOC50	—	—	—	CTX0	—	SCI3TxD	—	—	—
P10_2	IO	TIF2A	TOE01	—	—	TIOC51	TIOC51	—	—	CRX1	—	—	—	—	—	—
P10_3	IO	TIF3	TOE02	—	—	TIOC52	TIOC52	—	—	—	CTX1	—	—	—	—	—
P10_4	IO	TIF4	TOE03	—	—	TIOC53	TIOC53	—	—	CRX2	—	SCI1RxD	—	—	—	—

Table 2.56 List of Pin Alternative Functions in E1M-S Port Group 11

Port	General Input/ Output Port	1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative		Dedicated Function
		ALT_IN1	ALT_OUT1	ALT_IN2	ALT_OUT2	ALT_IN3	ALT_OUT3	ALT_IN4	ALT_OUT4	ALT_IN5	ALT_OUT5	ALT_IN6	ALT_OUT6	ALT_IN7	ALT_OUT7	
P11_0	IO	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
P11_1	IO	TIF13	—	—	TOD22B	—	APA0	—	TOE33	—	—	—	—	—	—	—
P11_2	IO	—	TOD80A	—	TOD80B	—	APA1	—	TOE20	—	—	—	—	—	—	—
P11_3	IO	—	TOD81A	—	TOD81B	—	APA2	—	TOE21	—	—	—	—	—	—	—
P11_4	IO	—	TOD82A	—	TOD82B	—	APA3	—	TOE22	—	—	—	—	—	—	—
P11_5	IO	—	TOD83A	—	TOD83B	—	APA4	—	TOE23	—	—	—	—	—	—	—

Table 2.57 List of Pin Alternative Functions in E1M-S Port Group 13 (1/2)

Port	General Input/ Output Port	1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative		Dedicated Function
		ALT_IN1	ALT_OUT1	ALT_IN2	ALT_OUT2	ALT_IN3	ALT_OUT3	ALT_IN4	ALT_OUT4	ALT_IN5	ALT_OUT5	ALT_IN6	ALT_OUT6	ALT_IN7	ALT_OUT7	
P13_0	IO	—	—	—	—	—	—	—	—	—	RHSB0CSD0	CSIH1TSI	—	—	—	—
P13_1	IO	—	—	—	—	—	—	—	—	—	RHSB0CSD1	—	CSIH1TSO	—	—	—
P13_2	IO	—	—	—	—	—	—	—	—	RHSB0SI0	—	CSIH1TSCK	CSIH1TSCK	—	—	—

Table 2.57 List of Pin Alternative Functions in E1M-S Port Group 13 (2/2)

Port	General Input/ Output Port	1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative		Dedicated Function
		ALT_IN1	ALT_OUT1	ALT_IN2	ALT_OUT2	ALT_IN3	ALT_OUT3	ALT_IN4	ALT_OUT4	ALT_IN5	ALT_OUT5	ALT_IN6	ALT_OUT6	ALT_IN7	ALT_OUT7	
P13_3	IO	—	—	—	—	—	—	—	—	RHSB0SI1	—	—	CSIH1TCSS0	—	—	—
P13_4	IO	—	—	—	—	—	—	—	—	RHSB0EMRG	—	—	CSIH1TCSS1	—	$\overline{\text{ERROROUT\_C}}$	—
P13_5	IO	—	—	—	—	—	—	—	—	—	RHSB1CSD0	—	—	—	—	—
P13_6	IO	—	—	—	—	—	—	—	—	—	RHSB1CSD1	—	—	—	—	—
P13_7	IO	—	—	—	—	—	—	—	—	RHSB1SI0	—	—	—	—	—	—
P13_8	IO	—	—	—	—	—	—	—	—	RHSB1SI1	—	—	—	—	—	—
P13_9	IO	—	—	—	—	—	—	—	—	RHSB1EMRG	—	—	—	—	—	—

Table 2.58 List of Pin Alternative Functions in E1M-S Port Group 14

Port	General Input/ Output Port	1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative		Dedicated Function	
		ALT_IN1	ALT_OUT1	ALT_IN2	ALT_OUT2	ALT_IN3	ALT_OUT3	ALT_IN4	ALT_OUT4	ALT_IN5	ALT_OUT5	ALT_IN6	ALT_OUT6	ALT_IN7	ALT_OUT7		
P14_0	I	TIF0A	—	—	—	—	—	—	—	—	—	—	—	—	—	AN020	/DSAN4P
P14_1	I	TIF1A	—	—	—	—	—	—	—	—	—	—	—	—	—	AN021	/DSAN4N
P14_2	I	TIF2A	—	—	—	—	—	—	—	—	—	—	—	—	—	AN022	/DSAN5P
P14_3	I	TIF0B	—	TIA00	—	—	—	—	—	—	—	—	—	—	—	AN023	/DSAN5N
P14_4	I	TIF1B	—	TIA01	—	—	—	—	—	—	—	—	—	—	—	AN030	/DSAN6P
P14_5	I	TIF2B	—	TIA02	—	—	—	—	—	—	—	—	—	—	—	AN031	/DSAN6N
P14_6	I	TIF3	—	—	—	—	—	—	—	—	—	—	—	—	—	AN032	/DSAN7P
P14_7	I	TIF4	—	TIA03	—	—	—	—	—	—	—	—	—	—	—	AN033	/DSAN7N
P14_8	I	TIF5	—	TIA04	—	—	—	—	—	—	—	—	—	—	—	AN040	
P14_9	I	TIF6	—	TIA05	—	—	—	—	—	—	—	—	—	—	—	AN041	
P14_10	I	TIF7	—	TIJ2	—	—	—	—	—	—	—	—	—	—	—	AN042	
P14_11	I	TIF8	—	TIJ3	—	—	—	—	—	—	—	—	—	—	—	AN043	
P14_12	I	TIF9	—	—	—	—	—	—	—	—	—	—	—	—	—	AN010	/DSAN2P
P14_13	I	TIF10	—	TIJ0	—	—	—	—	—	—	—	—	—	—	—	AN011	/DSAN2N
P14_14	I	TIF11	—	—	—	—	—	—	—	—	—	—	—	—	—	AN012	/DSAN3P
P14_15	I	TIF12	—	TIJ1	—	—	—	—	—	—	—	—	—	—	—	AN013	/DSAN3N

Table 2.59 List of Pin Alternative Functions in E1M-S Port Group 15

Port	General Input/ Output Port	1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative		Dedicated Function
		ALT_IN1	ALT_OUT1	ALT_IN2	ALT_OUT2	ALT_IN3	ALT_OUT3	ALT_IN4	ALT_OUT4	ALT_IN5	ALT_OUT5	ALT_IN6	ALT_OUT6	ALT_IN7	ALT_OUT7	
P15_0	I	TIF9	—	TIA06	—	—	—	—	—	—	—	—	—	—	—	AN050
P15_1	I	TIF10	—	—	—	—	—	—	—	—	—	—	—	—	—	AN051
P15_2	I	TIF11	—	—	—	—	—	—	—	—	—	—	—	—	—	AN052
P15_3	I	TIF12	—	—	—	—	—	—	—	—	—	—	—	—	—	AN053

Table 2.60 List of Pin Alternative Functions in E1M-S Port Group 16

Port	General Input/ Output Port	1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative		Dedicated Function
		ALT_IN1	ALT_OUT1	ALT_IN2	ALT_OUT2	ALT_IN3	ALT_OUT3	ALT_IN4	ALT_OUT4	ALT_IN5	ALT_OUT5	ALT_IN6	ALT_OUT6	ALT_IN7	ALT_OUT7	
P16_0	I	—	—	—	—	—	—	—	—	—	—	—	—	—	—	AN150
P16_1	I	—	—	—	—	—	—	—	—	—	—	—	—	—	—	AN151
P16_2	I	—	—	—	—	—	—	—	—	—	—	—	—	—	—	AN160
P16_3	I	—	—	—	—	—	—	—	—	—	—	—	—	—	—	AN161
P16_4	I	—	—	—	—	—	—	—	—	—	—	—	—	—	—	AN170
P16_5	I	—	—	—	—	—	—	—	—	—	—	—	—	—	—	AN171
P16_6	I	—	—	—	—	—	—	—	—	—	—	—	—	—	—	AN152
P16_7	I	—	—	—	—	—	—	—	—	—	—	—	—	—	—	AN162
P16_8	I	—	—	—	—	—	—	—	—	—	—	—	—	—	—	AN172

Table 2.61 List of Pin Alternative Functions in E1M-S Port Group 17

Port	General Input/ Output Port	1st Alternative		2nd Alternative		3rd Alternative		4th Alternative		5th Alternative		6th Alternative		7th Alternative		Dedicated Function
		ALT_IN1	ALT_OUT1	ALT_IN2	ALT_OUT2	ALT_IN3	ALT_OUT3	ALT_IN4	ALT_OUT4	ALT_IN5	ALT_OUT5	ALT_IN6	ALT_OUT6	ALT_IN7	ALT_OUT7	
P17_0	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RHSB0SOP, RHSB0SON
P17_1	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RHSB0FCLP, RHSB0FCLN
P17_2	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RHSB1SOP, RHSB1SON
P17_3	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RHSB1FCLP, RHSB1FCLN

2.3 DNF

Digital Noise Filter (DNF) eliminates digital noise from external input signals. This product includes two sorts of DNF: peripheral function DNF and edge detection DNF.

2.3.1 Example of Noise Elimination

Figure 2.7 shows an example of noise elimination in peripheral function DNF and edge detection DNF. In this example, the sampling clock, the sampling count, and the current output level are set to 1/2 of the DNF input clock, two (twice), and low, respectively. "○" in the figure means that high level is detected.

In input examples 1, 2, and 3, the output level changes from low to high because the same level is detected twice in a row through sampling. In input examples 4, 5, and 6, on the other hand, the same level is not detected twice consecutively. Therefore, these inputs are regarded as noise and the input signal state is eliminated.

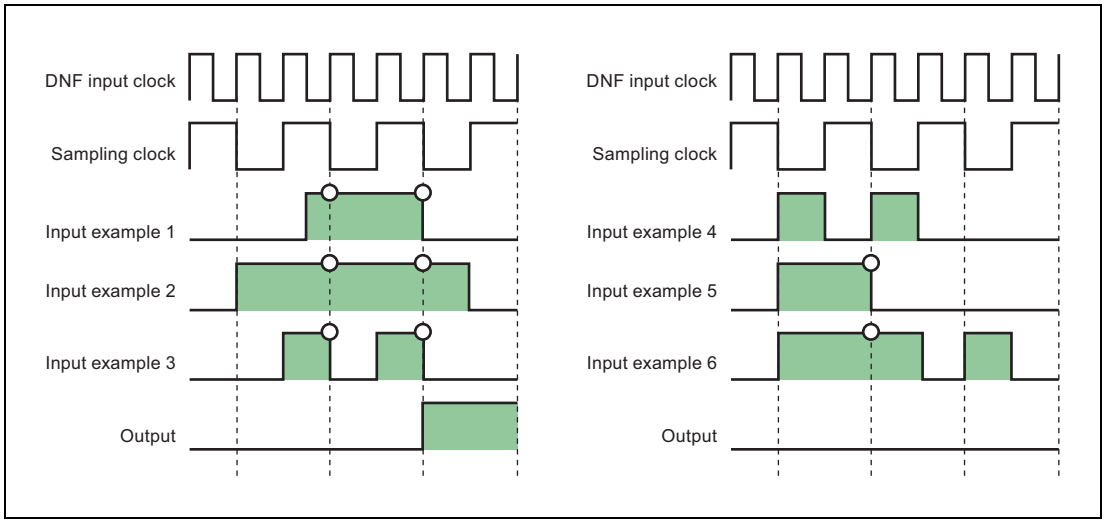


Figure 2.7 Timing Chart of Digital Noise Elimination

2.3.2 Peripheral Function DNF

2.3.2.1 Overview of Peripheral Function DNF

This DNF eliminates noise from the input function pins for the peripheral function.

Peripheral Function DNF has the following functions:

- Eliminates digital noise from input signals and outputs noiseless signals.
- Selects whether to output signals from which digital noise is eliminated or signals containing digital noise.
- Selects the digital noise elimination width from 2, 3, 4, and 5 counts of the sampling clock.
- Selects five types of sampling frequency shown below:
  1/1, 1/2, 1/4, 1/8, and 1/16 of the DNF input clock.
- The conditions for noise elimination can be set by each channel via the registers.
- The DNF input clock of DNF group number 0 is a low-speed peripheral clock.
- The DNF input clock of DNF group number 1 is an unmodulated high-speed peripheral clock.

2.3.2.2 Details of the Control Registers

Table 2.62 shows base addresses of peripheral function DNF. “n” indicates a number of the DNF group.

Table 2.62 Base Addresses of Peripheral Function DNF

DNFn	<DNFn_base> address
DNF0	FFC3 0000
DNF1	FFC3 0100

2.3.2.3 DNFP01nCTLm — Digital Noise Elimination Control Register

This register sets conditions for noise elimination of channel number m in DNF group number n.

Access: This register is readable/writable in 8-bit units.

Address: <DNFn_base> + 4_H × m (m: channel number)

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	NFEN	SLST[1:0]		—	—	PRS[2:0]		
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W

Table 2.63 DNFP01nCTLm Register Contents

Bit Position	Bit Name	Function
7	NFEN	Enables/disables digital noise elimination. 0: Digital noise is not eliminated. 1: Digital noise is eliminated.
6, 5	SLST[1:0]	Specifies the sampling count for digital noise elimination. 00: Sampling count = 2 01: Sampling count = 3 10: Sampling count = 4 11: Sampling count = 5
4, 3	—	Reserved. The read values are always 0. The write values should always be 0.
2 to 0	PRS[2:0]	Specifies the sampling clock division ratio for digital noise elimination. 000: DNF input clock / 1 001: DNF input clock / 2 010: DNF input clock / 4 011: DNF input clock / 8 100: DNF input clock / 16 Other than the above: Setting is prohibited.

2.3.2.4 Setting Procedures of Peripheral Function DNF

The following shows the procedures to set peripheral function DNF. If the input level to a pin varies during (4) and (5), an unexpected signal may be input to the peripheral function. Therefore, the corresponding flag at the peripheral function side or the like should be cleared in (6).

(1) Set [2:0]PRS2-0 and [6:5]SLST1-0 in the DNFP01nCTLm register.

(2) Set [7]NFEN in the DNFP01nCTLm register. This setting can be made at the same time as (1).

(3) Set a port register to select an alternative function.

(4) Wait for the following time: sampling clock period × number of samples + DNF input clock period × 2.

(5) Enable an operation of a peripheral function in the destination to which DNF is connected.

(6) Clear the flag of peripheral function, and the like.

To change the setting while the DNF is in operation, clear the NFEN bit (bit 7) in the DNFP01nCTLm register first, and then execute the procedure in steps (1) to (6) above to remake the settings.

2.3.2.5 Peripheral Function Pin intended for DNF insertion

The signals shown in **Table 2.64** are the targets of DNF insertion. In the table below, " √ " indicates with a support function and " — " without. For the registers without support functions, the settings should not be modified (the settings should be fixed to their values after reset).

Table 2.64 DNF Insertion Targets

DNF Group Number n	DNF Channel Number m	DNF Insertion Target Pin			E1M-S	
		Pin Name	Pin Function	Peripheral Function	BGA304	BGA252
0	0	IRQ0	External interrupt input	INTC	√	√
	1	IRQ1	External interrupt input	INTC	√	√
	2	IRQ2	External interrupt input	INTC	√	√
	3	IRQ3	External interrupt input	INTC	√	√
	4	IRQ4	External interrupt input	INTC	√	√
	5	IRQ5	External interrupt input	INTC	√	√
	6	IRQ6	External interrupt input	INTC	√	√
	7	IRQ7	External interrupt input	INTC	√	√
	10	ESO2	Emergency Hiz request input (for TSG20 PWM)	TAPA2 (via PIC)	√	√
	11	ESO3	Emergency Hiz request input (for TSG21 PWM)	TAPA3 (via PIC)	√	√
1	0	ADTRG0	SAR-AD conversion startup trigger input	SAR-AD0	√	√
	1	ADTRG1	SAR-AD conversion startup trigger input	SAR-AD1	√	√
	2	TAPTS00	Hall sensor input	TSG20	√	√
	3	TAPTS01	Hall sensor input	TSG20	√	√
	4	TAPTS02	Hall sensor input	TSG20	√	√
	5	TAPTS10	Hall sensor input	TSG21	√	√
	6	TAPTS11	Hall sensor input	TSG21	√	√
	7	TAPTS12	Hall sensor input	TSG21	√	√
	8	RHSB0EMRG	Emergency signal input	RHSB0	√	√
	9	RHSB1EMRG	Emergency signal input	RHSB1	√	√

2.3.3 Edge Detection DNF

2.3.3.1 Overview of the Function

This DNF detects an effective edge of a pin input signal. The signals to be used for edge detection can be selected either before or after noise elimination.

Edge detection DNF has the following functions:

- Eliminates digital noise from input signals and outputs noiseless signals.
- Selects whether to output signals from which digital noise is eliminated or signals containing digital noise.
- Selects the digital noise elimination width from 2, 3, 4, and 5 counts of the sampling clock.
- Selects five types of sampling frequency shown below:
  1/1, 1/2, 1/4, 1/8, and 1/16 of the DNF input clock.
- The conditions for noise elimination can be set collectively by each DNF group.
- The DNF input clock is a low-speed peripheral clock.

2.3.3.2 Details of the Control Registers

Table 2.65 shows base addresses of edge detection DNF. “n” indicates a number of the DNF group.

Table 2.65 Base Addresses of Edge Detection DNF

		E1M-S	
DNFn	**<DNFn_base> address**	**BGA304**	**BGA252**
DNF2	FFC3 0200	√	√
DNF3	FFC3 0300	√	√
DNF4	FFC3 0400	√	√
DNF5	FFC3 0500	√	—
DNF6	FFC3 0600	√	√
DNF7	FFC3 0700	√	√

2.3.3.3 DNFP02nCTL — Digital Noise Elimination Control Register

This register sets noise elimination conditions which are commonly used for all channels in DNF group number n.

Access: This register is readable/writable in 8-bit units.

Address: <DNFn_base> + 00_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	SLST[1:0]		—	—	PRS[2:0]		
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R	R	R/W	R/W	R/W

Table 2.66 DNFP02nCTL Register Contents

Bit Position	Bit Name	Function
7	—	Reserved. The read value is always 0. The write value should always be 0.
6, 5	SLST[1:0]	Specifies the sampling count for digital noise elimination. 00: Sampling count = 2 01: Sampling count = 3 10: Sampling count = 4 11: Sampling count = 5
4, 3	—	Reserved. The read values are always 0. The write values should always be 0.
2 to 0	PRS[2:0]	Specifies the sampling clock division ratio for digital noise elimination. 000: DNF input clock / 1 001: DNF input clock / 2 010: DNF input clock / 4 011: DNF input clock / 8 100: DNF input clock / 16 Other than the above: Setting is prohibited.

2.3.3.4 DNFP02nEDCm — Edge Detection Control Register

This register controls edge detection of channel number m in DNF group number n.

Access: This register is readable/writable in 8-bit units.

Address: <DNFn_base> + 4_H + 8_H × m (m: channel number)

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	NFEN	—	—	DMD[1:0]		—	—	—
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R/W	R/W	R	R	R

Table 2.67 DNFP02nEDCm Register Contents

Bit Position	Bit Name	Function
7	NFEN	Enables/disables digital noise elimination. 0: Edge is detected for signals containing digital noise. 1: Edge is detected for signals from which digital noise is eliminated.
6, 5	—	Reserved. The read values are always 0. The write values should always be 0.
4, 3	DMD[1:0]	Enables/disables edge detection and sets edge to be detected 00: Edge detection is disabled. 01: Rising edge is detected. 10: Falling edge is detected. 11: Both edges are detected.
2 to 0	—	Reserved. The read values are always 0. The write values should always be 0.

2.3.3.5 DNFP02nEDFm — Edge Detection Flag Register

This register has an edge detection flag and a flag clear bit of channel number m in DNF group number n.

Access: This register is readable/writable in 8-bit units.

Address: <DNFn_base> + 8_H + 8_H × m (m: channel number)

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	EDF	—	—	—	—	—	—	CLED
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 2.68 DNFP02nEDFm Register Contents

Bit Position	Bit Name	Function
7	EDF	This bit holds a result of edge detection based on the edge detection setting for channel number m. This bit is cleared by writing 1 to the CLED bit and continues to hold 1 unless it is cleared. Its state does not change while 1 is held, even if an edge is detected. 0: Effective edge has not been detected. 1: Effective edge has been detected.
6 to 1	—	Reserved. The read values are always 0. The write values should always be 0.
0	CLED	Writing 1 to this bit clears the status flag of EDF, bit 7. In case of contention between the writing of 1 and a setting source of bit 7, EDF clearing takes priority. Even writing 0 cannot change the state. The read value is always 0.

2.3.3.6 Setting Procedures of Edge Detection DNF

The following shows the procedures to set edge detection DNF. When edge detection is performed for signals containing digital noise, the steps with (*) can be skipped. If the input level to a pin varies during (4) and (5), an unexpected signal may be input to the peripheral function. Therefore, the corresponding flag at the peripheral function side and the like should be cleared in (6).

(1) Set [2:0]PRS2-0 and [6:5]SLST1-0 in the DNFP02nCTL register.(*)

(2) Set [7]NFEN in the DNFP02nEDCm register.

(3) Set a port register to general input port mode.

(4) Wait for the following time: sampling clock period × number of samples + DNF input clock period × 2. (*)

(5) Enable edge detection in [4:3]DMD1-0 of the DNFP02nEDCm register.

(6) Clear an edge detection flag in [0]CLED of the DNFP02nEDFm register.

Set [4:3]DMD1-0 in the DNFP02nEDCm register to 00B, and then proceed the setting again in the order from steps (1) to (6), if the setting is to be changed while edge detection DNF is under the operation.

2.3.3.7 Pn_m Pins Intended for DNF Insertion

The pins shown in **Table 2.69** are the targets of DNF insertion. In the table below, " √ " indicates with a support function and " — " without. For the registers without support functions, the settings should not be modified (the settings should be fixed to their values after reset).

Table 2.69 DNF Insertion Targets (1/2)

DNF Group Number n	DNF Channel Number m	Pin Name	E1M-S	
			BGA304	BGA252
2	0	P2_0	√	√
	1	P2_1	√	√
	2	P2_2	√	√
	3	P2_3	√	√
	4	P2_4	√	√
	5	P2_5	√	√
	6	P2_6	√	√
	7	P2_7	√	√
3	0	P3_0	√	√
	1	P3_1	√	√
	2	P3_2	√	√
	3	P3_3	√	√
	4	P3_4	√	√
	5	P3_5	√	√
	6	P3_6	√	√
	7	P3_7	√	√
4	0	P4_0	√	√
	1	P4_1	√	√
	2	P4_2	√	√
	3	P4_3	√	√
	4	P4_4	√	√
	5	P4_5	√	√
	6	P4_6	√	√
	7	P4_7	√	√
	8	P4_8	√	√
	9	P4_9	√	√
	10	P4_10	√	√
	11	P4_11	√	√
	12	P4_12	√	√
	13	P4_13	√	√
	14	P4_14	√	√
	15	P4_15	√	√

Table 2.69 DNF Insertion Targets (2/2)

DNF Group Number n	DNF Channel Number m	Pin Name	E1M-S	
			BGA304	BGA252
5	0	P7_0	√	—
	1	P7_1	√	—
	2	P7_2	√	—
	3	P7_3	√	—
	4	P7_4	√	—
	5	P7_5	√	—
	6	P7_6	√	—
	7	P7_7	√	—
	8	P7_8	√	—
	9	P7_9	√	—
	10	P7_10	√	—
	11	P7_11	√	—
6	0	P14_0	√	√
	1	P14_1	√	√
	2	P14_2	√	√
	3	P14_3	√	√
	4	P14_4	√	√
	5	P14_5	√	√
	6	P14_6	√	√
	7	P14_7	√	√
	8	P14_8	√	√
	9	P14_9	√	√
	10	P14_10	√	√
	11	P14_11	√	√
	12	P14_12	√	√
	13	P14_13	√	√
	14	P14_14	√	√
	15	P14_15	√	√
7	0	P16_0	√	√
	1	P16_1	√	√
	2	P16_2	√	√
	3	P16_3	√	√
	4	P16_4	√	√
	5	P16_5	√	√
	6	P16_6	√	√
	7	P16_7	√	√
	8	P16_8	√	√

2.4 POD Control

2.4.1 Overview of the Function

The output buffer of the related pins shown in **Table 2.71** can be controlled according to the $\overline{\text{POD}}$ (port output disable) pin input level. This function is always available when target pins are set to output regardless of the selected function. The output buffer is controlled by the POD function independently of the bus cycle. POD control takes precedence over output settings of a port register and output designation of each IP, but its priority is lower than reset.

Table 2.70 State of the $\overline{\text{POD}}$ Pin

$\overline{\text{POD}}$ Pin State	Description
0	Pin output is not effective (high impedance).
1	Pin output is effective (each setting function).

Table 2.71 List of POD Control Target Pins (1/3)

Port Group	Pin Name (Value after Reset)	E1M-S	
		BGA304	BGA252
P0	P0_11	√	√
	P0_12	√	√
	P0_13	√	√
	P0_14	√	√
P1	P1_0	√	√
	P1_1	√	√
	P1_2	√	√
	P1_3	√	√
	P1_4	√	√
	P1_5	√	√
	P1_6	√	√
	P1_7	√	√
	P1_8	√	√
	P1_9	√	√
	P1_10	√	√
	P1_11	√	√
	P1_12	√	√
	P1_13	√	√
	P1_14	√	√
	P1_15	√	√
P2	P2_0	√	√
	P2_1	√	√
	P2_2	√	√
	P2_3	√	√
	P2_4	√	√
	P2_5	√	√
	P2_6	√	√
	P2_7	√	√

Table 2.71 List of POD Control Target Pins (2/3)

Port Group	Pin Name (Value after Reset)	E1M-S	
		BGA304	BGA252
P3	P3_0	√	√
	P3_1	√	√
	P3_2	√	√
	P3_3	√	√
	P3_4	√	√
	P3_5	√	√
	P3_6	√	√
	P3_7	√	√
P4	P4_0	√	√
	P4_1	√	√
	P4_2	√	√
	P4_3	√	√
	P4_4	√	√
	P4_5	√	√
	P4_6	√	√
	P4_7	√	√
	P4_8	√	√
	P4_9	√	√
	P4_10	√	√
	P4_11	√	√
	P4_12	√	√
	P4_13	√	√
	P4_14	√	√
	P4_15	√	√
P5	P5_0	√	√
	P5_1	√	√
	P5_3	√	√
	P5_4	√	√
	P5_5	√	√
	P5_6	√	√
	P5_7	√	√
	P5_8	√	√
	P5_9	√	√

Table 2.71 List of POD Control Target Pins (3/3)

Port Group	Pin Name (Value after Reset)	E1M-S	
		BGA304	BGA252
P6	P6_0	√	√
	P6_1	√	√
	P6_2	√	√
	P6_3	√	√
	P6_4	√	√
	P6_5	√	√
	P6_6	√	√
	P6_7	√	√
	P6_8	√	√
	P6_9	√	√
	P6_10	√	√
	P6_11	√	√
P7	P7_0	√	—
	P7_1	√	—
	P7_2	√	—
	P7_3	√	—
	P7_4	√	—
	P7_5	√	—
	P7_6	√	—
	P7_7	√	—
	P7_8	√	—
	P7_9	√	—
	P7_10	√	—
	P7_11	√	—
P8	P8_0	√	—
	P8_1	√	—
	P8_2	√	—
	P8_3	√	—
	P8_4	√	—
	P8_5	√	—
	P8_6	√	—
	P8_7	√	—
P10	P10_1	√	—
	P10_2	√	—
	P10_3	√	—
	P10_4	√	—
P11	P11_1	√	√
	P11_2	√	√
	P11_3	√	√
	P11_4	√	√
	P11_5	√	√

2.5 Pin Description

2.5.1 Overview

This subsection describes pin functions, external pin lists, and external pin states at a reset and in other each operating status.

2.5.2 List of Pin Functions

Table 2.72 lists functions of each pin. " √ " indicates with a support function and " — " without. Do not use the pins without support functions.

Table 2.72 Pin Function (1/15)

Category	Pin Name	IO	Function	E1M-S BGA304	E1M-S BGA252
ATU-IV (Common area)	TCLKA	I	External clock input	√	√
	TCLKB	I	External clock input	√	√
ATU-IV (Timer A)	TIA00	I	Input capture trigger to each channel of Timer A	√	√
	TIA01	I		√	√
	TIA02	I		√	√
	TIA03	I		√	√
	TIA04	I		√	√
	TIA05	I		√	√
	TIA06	I		√	√
ATU-IV (Timer C)	TIOC00	IO	Input capture trigger and output compare output to Timer C	√	√
	TIOC01	IO		√	√
	TIOC02	IO		√	√
	TIOC03	IO		√	√
	TIOC10	IO		√	√
	TIOC11	IO		√	√
	TIOC12	IO		√	√
	TIOC13	IO		√	√
	TIOC20	IO		√	√
	TIOC21	IO		√	√
	TIOC22	IO		√	√
	TIOC23	IO		√	√
	TIOC30	IO		√	√
	TIOC31	IO		√	√
	TIOC32	IO		√	√
	TIOC33	IO		√	√
	TIOC40	IO		√	√
	TIOC41	IO		√	√
	TIOC42	IO		√	√
	TIOC43	IO		√	√

Table 2.72 Pin Function (2/15)

Category	Pin Name	IO	Function	E1M-S BGA304	E1M-S BGA252
ATU-IV (Timer C)	TIOC50	IO	Input capture trigger, and output compare output of Timer C	√	√
	TIOC51	IO		√	√
	TIOC52	IO		√	√
	TIOC53	IO		√	√
	TIOC60	IO		√	√
	TIOC61	IO		√	√
	TIOC62	IO		√	√
	TIOC63	IO		√	√
	TIOC70	IO		√	√
	TIOC71	IO		√	√
	TIOC72	IO		√	√
	TIOC73	IO		√	√
ATU-IV (Timer D)	TOD00A	O	One-shot pulse output of Timer D	√	√
	TOD01A	O		√	√
	TOD02A	O		√	√
	TOD03A	O		√	√
	TOD10A	O		√	√
	TOD11A	O		√	√
	TOD12A	O		√	√
	TOD13A	O		√	√
	TOD20A	O		√	√
	TOD21A	O		√	√
	TOD22A	O		√	√
	TOD23A	O		√	√
	TOD30A	O		√	√
	TOD31A	O		√	√
	TOD32A	O		√	√
	TOD33A	O		√	√
	TOD40A	O		√	√
	TOD41A	O		√	√
	TOD42A	O		√	√
	TOD43A	O		√	√
	TOD50A	O		√	√
	TOD51A	O		√	√
	TOD52A	O		√	√
	TOD53A	O		√	√
	TOD60A	O		√	√
	TOD61A	O		√	√
	TOD62A	O		√	√
	TOD63A	O		√	√

Table 2.72 Pin Function (3/15)

Category	Pin Name	IO	Function	E1M-S BGA304	E1M-S BGA252
ATU-IV (Timer D)	TOD70A	O	One-shot pulse output of Timer D	√	√
	TOD71A	O		√	√
	TOD72A	O		√	√
	TOD73A	O		√	√
	TOD80A	O		√	√
	TOD81A	O		√	√
	TOD82A	O		√	√
	TOD83A	O		√	√
	TOD00B	O		√	√
	TOD01B	O		√	√
	TOD02B	O		√	√
	TOD03B	O		√	√
	TOD10B	O		√	√
	TOD11B	O		√	√
	TOD12B	O		√	√
	TOD13B	O		√	√
	TOD20B	O		√	√
	TOD21B	O		√	√
	TOD22B	O		√	√
	TOD23B	O		√	√
	TOD30B	O		√	√
	TOD31B	O		√	√
	TOD32B	O		√	√
	TOD33B	O		√	√
	TOD40B	O		√	√
	TOD41B	O		√	√
	TOD42B	O		√	√
	TOD43B	O		√	√
	TOD50B	O		√	√
	TOD51B	O		√	√
	TOD52B	O		√	√
	TOD53B	O		√	√
	TOD60B	O		√	√
	TOD61B	O		√	√
	TOD62B	O		√	√
	TOD63B	O		√	√
	TOD70B	O		√	√
	TOD71B	O		√	√
	TOD72B	O		√	√
	TOD73B	O		√	√

Table 2.72 Pin Function (4/15)

Category	Pin Name	IO	Function	E1M-S BGA304	E1M-S BGA252
ATU-IV (Timer D)	TOD80B	O	PWM output of Timer D	√	√
	TOD81B	O		√	√
	TOD82B	O		√	√
	TOD83B	O		√	√
ATU-IV (Timer E)	TOE00	O	PWM output of Timer E	√	√
	TOE01	O		√	√
	TOE02	O		√	√
	TOE03	O		√	√
	TOE10	O		√	√
	TOE11	O		√	√
	TOE12	O		√	√
	TOE13	O		√	√
	TOE20	O		√	√
	TOE21	O		√	√
	TOE22	O		√	√
	TOE23	O		√	√
	TOE30	O		√	√
	TOE31	O		√	√
	TOE32	O		√	√
	TOE33	O		√	√
	TOE40	O		√	√
	TOE41	O		√	√
	TOE42	O		√	√
	TOE43	O		√	√
	TOE50	O		√	√
	TOE51	O		√	√
	TOE52	O		√	√
	TOE53	O		√	√
	TOE60	O		√	√
	TOE61	O		√	√
	TOE62	O		√	√
	TOE63	O		√	√
ATU-IV (Timer F)	TIF0A	I	Event input signal to Timer F	√	√
	TIF1A	I		√	√
	TIF2A	I		√	√
	TIF0B	I		√	√
	TIF1B	I		√	√
	TIF2B	I		√	√
	TIF3	I		√	√
	TIF4	I		√	√
	TIF5	I		√	√

Table 2.72 Pin Function (5/15)

Category	Pin Name	IO	Function	E1M-S BGA304	E1M-S BGA252
ATU-IV (Timer F)	TIF6	I	Event input signal to Timer F	√	√
	TIF7	I		√	√
	TIF8	I		√	√
	TIF9	I		√	√
	TIF10	I		√	√
	TIF11	I		√	√
	TIF12	I		√	√
	TIF13	I		√	√
	TIF14	I		√	√
	TIF15	I		√	√
ATU-IV (Timer J)	TIJ0	I	Input capture trigger of Timer J	√	√
	TIJ1	I		√	√
	TIJ2	I		√	√
	TIJ3	I		√	√
	TIJ4	I		√	√
	TIJ5	I		√	√
APA	APA0	O	APA output	√	√
	APA1	O		√	√
	APA2	O		√	√
	APA3	O		√	√
	APA4	O		√	√
	APA5	O		√	√
	APA6	O		√	√
	APA7	O		√	√
	APA8	O		√	√
	APA9	O		√	√
	APA10	O		√	√
	APA11	O		√	√
	APA12	O		√	√
	APA13	O		√	√
	APA14	O		√	√
	APA15	O		√	√
TSG20	TSO00	O	Timer up/down status output	√	√
	TSO01	O	Timer PWM output	√	√
	TSO02	O	**CAUTION**	√	√
	TSO03	O	Output from PIC should be used.	√	√
	TSO04	O		√	√
	TSO05	O		√	√
	TSO06	O		√	√
	TSO07	O	A/D trigger diagnosis output	√	√
	TAPTS00	I	Hall sensor input	√	√
	TAPTS01	I		√	√
	TAPTS02	I		√	√

Table 2.72 Pin Function (6/15)

Category	Pin Name	IO	Function	E1M-S BGA304	E1M-S BGA252
TSG21	TSO10	O	Timer up/down status output	√	√
	TSO11	O	Timer PWM output	√	√
	TSO12	O	**CAUTION**	√	√
	TSO13	O	Output from PIC should be used.	√	√
	TSO14	O		√	√
	TSO15	O		√	√
	TSO16	O		√	√
	TSO17	O	A/D trigger diagnosis output	√	√
	TAPTS10	I	Hall sensor input	√	√
	TAPTS11	I		√	√
	TAPTS12	I		√	√
TAPA (via PIC)	ESO2	I	Emergency Hiz request input (for TSG20 PWM)	√	√
	ESO3	I	Emergency Hiz request input (for TSG21 PWM)	√	√
LIN0	LRX0	I	LIN0 receive data input	√	√
	LTX0	O	LIN0 transmit data output	√	√
SCI0	SCI0TxD	O	SCI0 transmit data output	√	√
	SCI0RxD	I	SCI0 receive data input	√	√
	SCI0SCK	IO	SCI0 serial clock input/output	√	√
SCI1	SCI1TxD	O	SCI1 transmit data output	√	√
	SCI1RxD	I	SCI1 receive data input	√	√
	SCI1SCK	IO	SCI1 serial clock input/output	√	√
SCI2	SCI2TxD	O	SCI2 transmit data output	√	√
	SCI2RxD	I	SCI2 receive data input	√	√
	SCI2SCK	IO	SCI2 serial clock input/output	√	√
SCI3	SCI3TxD	O	SCI3 transmit data output	√	√
	SCI3RxD	I	SCI3 receive data input	√	√
	SCI3SCK	IO	SCI3 serial clock input/output	√	√
CSIH0	CSIH0TSI	I	CSIH0 receive data input	√	√
	CSIH0TSO	O	CSIH0 transmit data output	√	√
	CSIH0TSCK	IO	CSIH0 serial clock input/output	√	√
	$\overline{\text{CSIH0TSSI}}$	I	CSIH0 slave select signal input	√	√
	CSIH0TRY	IO	CSIH0 handshake signal input/output between master and slave	√	√
	CSIH0TCSS0	O	CSIH0 slave select signal output	√	√
	CSIH0TCSS1	O		√	√
	CSIH0TCSS2	O		√	√
	CSIH0TCSS3	O		√	√
	CSIH0TCSS4	O		√	√
	CSIH0TCSS5	O		√	√

Table 2.72 Pin Function (7/15)

Category	Pin Name	IO	Function	E1M-S BGA304	E1M-S BGA252
CSIH1	CSIH1TSI	I	CSIH1 receive data input	√	√
	CSIH1TSO	O	CSIH1 transmit data output	√	√
	CSIH1TSCK	IO	CSIH1 serial clock input/output	√	√
	$\overline{\text{CSIH1TSSI}}$	I	CSIH1 slave select signal input	√	√
	CSIH1TRY	IO	CSIH1 handshake signal input/output between master and slave	√	√
	CSIH1TCSS0	O	CSIH1 slave select signal output	√	√
	CSIH1TCSS1	O		√	√
	CSIH1TCSS2	O		√	√
	CSIH1TCSS3	O		√	√
CSIH2	CSIH2TSI	I	CSIH2 receive data input	√	√
	CSIH2TSO	O	CSIH2 transmit data output	√	√
	CSIH2TSCK	IO	CSIH2 serial clock input/output	√	√
	$\overline{\text{CSIH2TSSI}}$	I	CSIH2 slave select signal input	√	√
	CSIH2TRY	IO	CSIH2 handshake signal input/output between master and slave	√	√
	CSIH2TCSS0	O	CSIH2 slave select signal output	√	√
	CSIH2TCSS1	O		√	√
	CSIH2TCSS2	O		√	√
	CSIH2TCSS3	O		√	√
CSIH3	CSIH3TSI	I	CSIH3 receive data input	√	√
	CSIH3TSO	O	CSIH3 transmit data output	√	√
	CSIH3TSCK	IO	CSIH3 serial clock input/output	√	√
	$\overline{\text{CSIH3TSSI}}$	I	CSIH3 slave select signal input	√	√
	CSIH3TRY	IO	CSIH3 handshake signal input/output between master and slave	√	√
	CSIH3TCSS0	O	CSIH3 slave select signal output	√	√
	CSIH3TCSS1	O		√	√
	CSIH3TCSS2	O		√	√
	CSIH3TCSS3	O		√	√
RSCAN0	CTX0	O	CAN0 transmit data output	√	√
	CRX0	I	CAN0 receive data input	√	√
RSCAN1	CTX1	O	CAN1 transmit data output	√	√
	CRX1	I	CAN1 receive data input	√	√
RSCAN2	CTX2	O	CAN2 transmit data output	√	√
	CRX2	I	CAN2 receive data input	√	√
RSCAN3	CTX3	O	CAN3 transmit data output	√	√
	CRX3	I	CAN3 receive data input	√	√
FlexRay	FR0RXDA	I	FlexRay0 channel A receive data input	√	√
	FR0TXDA	O	FlexRay0 channel A transmit data output	√	√
	$\overline{\text{FR0ENA}}$	O	FlexRay0 channel A transmit data enable	√	√
	FR0RXDB	I	FlexRay0 channel B receive data input	√	√
	FR0TXDB	O	FlexRay0 channel B transmit data output	√	√
	$\overline{\text{FR0ENB}}$	O	FlexRay0 channel B transmit data enable	√	√

Table 2.72 Pin Function (8/15)

Category	Pin Name	IO	Function	E1M-S BGA304	E1M-S BGA252
RHSB0	RHSB0FCLP	O	RHSB0 differential clock output	√	√
	RHSB0FCLN	O		√	√
	RHSB0SOP	O	RHSB0 differential data output	√	√
	RHSB0SON	O		√	√
	RHSB0CSD0	O	RHSB0 chip select output	√	√
	RHSB0CSD1	O		√	√
	RHSB0SI0	I	RHSB0 serial input	√	√
	RHSB0SI1	I		√	√
	RHSB0EMRG	I	RHSB0 emergency signal input	√	√
RHSB1	RHSB1FCLP	O	RHSB1 differential clock output	√	√
	RHSB1FCLN	O		√	√
	RHSB1SOP	O	RHSB1 differential data output	√	√
	RHSB1SON	O		√	√
	RHSB1CSD0	O	RHSB1 chip select output	√	√
	RHSB1CSD1	O		√	√
	RHSB1SI0	I	RHSB1 serial input	√	√
	RHSB1SI1	I		√	√
	RHSB1EMRG	I	RHSB1 emergency signal input	√	√
ADC0	AN000	I	ADC0 analog input	√	√
	AN001	I		√	√
	AN002	I		√	√
	AN003	I		√	√
	AN010	I		√	√
	AN011	I		√	√
	AN012	I		√	√
	AN013	I		√	√
	AN020	I		√	√
	AN021	I		√	√
	AN022	I		√	√
	AN023	I		√	√
	AN030	I		√	√
	AN031	I		√	√
	AN032	I		√	√
	AN033	I		√	√
	AN040	I		√	√
	AN041	I		√	√
	AN042	I		√	√
	AN043	I		√	√
	AN050	I		√	√
	AN051	I		√	√
	AN052	I		√	√
	AN053	I		√	√

Table 2.72 Pin Function (9/15)

Category	Pin Name	IO	Function	E1M-S BGA304	E1M-S BGA252
ADC0	ADTRG0	I	ADC0 conversion startup trigger input	√	√
	ADEND0	O	ADC0 conversion timing monitor output	√	√
ADC1	AN100	I	ADC1 analog input	√	√
	AN101	I		√	√
	AN102	I		√	√
	AN110	I		√	√
	AN111	I		√	√
	AN112	I		√	√
	AN120	I		√	√
	AN121	I		√	√
	AN122	I		√	√
	AN130	I		√	√
	AN131	I		√	√
	AN132	I		√	√
	AN140	I		√	√
	AN141	I		√	√
	AN142	I		√	√
	AN150	I		√	√
	AN151	I		√	√
	AN152	I		√	√
	AN160	I		√	√
	AN161	I		√	√
	AN162	I		√	√
	AN170	I		√	√
	AN171	I		√	√
	AN172	I		√	√
	ADTRG1	I	ADC1 conversion startup trigger input	√	√
	ADEND1	O	ADC1 conversion timing monitor output	√	√
DSADC0	DSAN0P	I	DSADC0 analog input	√	√
	DSAN0N	I		√	√
	DSADTRG0	I	DSADC0 conversion startup trigger input	√	√
	DSADEND0	O	DSADC0 conversion timing monitor output	√	√
DSADC1	DSAN1P	I	DSADC1 analog input	√	√
	DSAN1N	I		√	√
	DSADTRG1	I	DSADC1 conversion startup trigger input	√	√
	DSADEND1	O	DSADC1 conversion timing monitor output	√	√
DSADC2	DSAN2P	I	DSADC2 analog input	√	√
	DSAN2N	I		√	√
	DSADTRG2	I	DSADC2 conversion startup trigger input	√	√
	DSADEND2	O	DSADC2 conversion timing monitor output	√	√

Table 2.72 Pin Function (10/15)

Category	Pin Name	IO	Function	E1M-S BGA304	E1M-S BGA252
DSADC3	DSAN3P	I	DSADC3 analog input	√	√
	DSAN3N	I		√	√
	DSADTRG3	I	DSADC3 conversion startup trigger input	√	√
	DSADEND3	O	DSADC3 conversion timing monitor output	√	√
DSADC4	DSAN4P	I	DSADC4 analog input	√	√
	DSAN4N	I		√	√
	DSADTRG4	I	DSADC4 conversion startup trigger input	√	√
	DSADEND4	O	DSADC4 conversion timing monitor output	√	√
DSADC5	DSAN5P	I	DSADC5 analog input	√	√
	DSAN5N	I		√	√
	DSADTRG5	I	DSADC5 conversion startup trigger input	√	√
	DSADEND5	O	DSADC5 conversion timing monitor output	√	√
DSADC6	DSAN6P	I	DSADC6 analog input	√	√
	DSAN6N	I		√	√
	DSADTRG6	I	DSADC6 conversion startup trigger input	√	√
	DSADEND6	O	DSADC6 conversion timing monitor output	√	√
DSADC7	DSAN7P	I	DSADC7 analog input	√	√
	DSAN7N	I		√	√
	DSADTRG7	I	DSADC7 conversion startup trigger input	√	√
	DSADEND7	O	DSADC7 conversion timing monitor output	√	√
Port	$\overline{\text{POD}}$	I	Port output disable	√	√
ECM	$\overline{\text{ERROROUT\_M}}$	O	Error output (Master)	√	√
	$\overline{\text{ERROROUT\_C}}$	O	Error output (Checker)	√	√
Interrupt	NMI	I	Non-maskable interrupt	√	√
	IRQ0	I	External interrupt input	√	√
	IRQ1	I		√	√
	IRQ2	I		√	√
	IRQ3	I		√	√
	IRQ4	I		√	√
	IRQ5	I		√	√
	IRQ6	I		√	√
	IRQ7	I		√	√
Clock	EXTAL	I	External clock	√	√
	XTAL	O	Crystal	√	√
	CK	O	Peripheral clock	√	√
Reset	$\overline{\text{RES}}$	I	External reset input	√	√
Mode setting	MD0	I	Mode setting	√	√
	MD1	I		√	√
	FLMODE	I		√	√

Table 2.72 Pin Function (11/15)

Category	Pin Name	IO	Function	E1M-S BGA304	E1M-S BGA252
Nexus/ JTAG	$\overline{\text{TRST}}$	I	Reset input	√	√
	TDO	O	Serial data output	√	√
	TMS	I	Mode select input	√	√
	TDI	I	Serial data input	√	√
	TCK	I	Clock input	√	√
	$\overline{\text{DRDY}}$	O	Ready output	√	√
	$\overline{\text{EVTO}}$	O	Event trigger output (exclusive use)	√	√
	$\overline{\text{EVTI}}$	I	Event trigger input	√	√
LPD	$\overline{\text{LPDRST}}$	I	LPD reset input	√	√
	LPDO	O	LPD4 pin data output	√	√
	LPDI	I	LPD4 pin data input	√	√
	LPDCLKO	O	LPD4 pin clock output	√	√
	LPDCLKI	I	LPD4 pin clock input	√	√
	$\overline{\text{EVTO}}$	O	Event trigger output (pin dedicated for TMS)	√	√
AUD	$\overline{\text{AUDRST}}$	I	AUD reset input	√	√
	AUDCK	I	External input clock	√	√
	$\overline{\text{AUDSYNC}}$	I	Data start position recognition signal input	√	√
	AUDATA3	IO	Monitor address input / data input/output	√	√
	AUDATA2	IO	Monitor address input / data input/output	√	√
	AUDATA1	IO	Monitor address input / data input/output	√	√
	AUDATA0	IO	Monitor address input / data input/output	√	√
FLSCI (writer I/F)	FLSCI3TX (FPDT)	O	Transmit data output	√	√
	FLSCI3RX (FPDR)	I	Receive data input	√	√
	FLSCI3SCK (FPCK)	I	Serial clock input	√	√
Port group 0	P0_0	IO	General input/output port	√	√
	P0_1	IO		√	√
	P0_2	IO		√	√
	P0_3	IO		√	√
	P0_4	IO		√	√
	P0_5	IO		√	√
	P0_6	IO		√	√
	P0_7	IO		√	√
	P0_8	IO		√	√
	P0_9	IO		√	√
	P0_10	IO		√	√
	P0_11	IO		√	√
	P0_12	IO		√	√
	P0_13	IO		√	√
	P0_14	IO		√	√

Table 2.72 Pin Function (12/15)

Category	Pin Name	IO	Function	E1M-S BGA304	E1M-S BGA252
Port group 1	P1_0	IO	General input/output port	√	√
	P1_1	IO		√	√
	P1_2	IO		√	√
	P1_3	IO		√	√
	P1_4	IO		√	√
	P1_5	IO		√	√
	P1_6	IO		√	√
	P1_7	IO		√	√
	P1_8	IO		√	√
	P1_9	IO		√	√
	P1_10	IO		√	√
	P1_11	IO		√	√
	P1_12	IO		√	√
	P1_13	IO		√	√
	P1_14	IO		√	√
	P1_15	IO		√	√
Port group 2	P2_0	IO	General input/output port with edge detection function	√	√
	P2_1	IO		√	√
	P2_2	IO		√	√
	P2_3	IO		√	√
	P2_4	IO		√	√
	P2_5	IO		√	√
	P2_6	IO		√	√
	P2_7	IO		√	√
Port group 3	P3_0	IO	General input/output port with edge detection function	√	√
	P3_1	IO		√	√
	P3_2	IO		√	√
	P3_3	IO		√	√
	P3_4	IO		√	√
	P3_5	IO		√	√
	P3_6	IO		√	√
	P3_7	IO		√	√
Port group 4	P4_0	IO	General input/output port with edge detection function	√	√
	P4_1	IO		√	√
	P4_2	IO		√	√
	P4_3	IO		√	√
	P4_4	IO		√	√
	P4_5	IO		√	√
	P4_6	IO		√	√
	P4_7	IO		√	√
	P4_8	IO		√	√
	P4_9	IO		√	√
	P4_10	IO		√	√

Table 2.72 Pin Function (13/15)

Category	Pin Name	IO	Function	E1M-S BGA304	E1M-S BGA252
Port group 4	P4_11	IO	General input/output port with edge detection function	√	√
	P4_12	IO		√	√
	P4_13	IO		√	√
	P4_14	IO		√	√
	P4_15	IO		√	√
Port group 5	P5_0	IO	General input/output port	√	√
	P5_1	IO		√	√
	P5_2	IO		√	√
	P5_3	IO		√	√
	P5_4	IO		√	√
	P5_5	IO		√	√
	P5_6	IO		√	√
	P5_7	IO		√	√
	P5_8	IO		√	√
	P5_9	IO		√	√
Port group 6	P6_0	IO	General input/output port	√	√
	P6_1	IO		√	√
	P6_2	IO		√	√
	P6_3	IO		√	√
	P6_4	IO		√	√
	P6_5	IO		√	√
	P6_6	IO		√	√
	P6_7	IO		√	√
	P6_8	IO		√	√
	P6_9	IO		√	√
	P6_10	IO		√	√
	P6_11	IO		√	√
Port group 7	P7_0	IO	General input/output port with edge detection function	√	—
	P7_1	IO		√	—
	P7_2	IO		√	—
	P7_3	IO		√	—
	P7_4	IO		√	—
	P7_5	IO		√	—
	P7_6	IO		√	—
	P7_7	IO		√	—
	P7_8	IO		√	—
	P7_9	IO		√	—
	P7_10	IO		√	—
	P7_11	IO		√	—

Table 2.72 Pin Function (14/15)

Category	Pin Name	IO	Function	E1M-S BGA304	E1M-S BGA252
Port group 8	P8_0	IO	General input/output port	√	—
	P8_1	IO		√	—
	P8_2	IO		√	—
	P8_3	IO		√	—
	P8_4	IO		√	—
	P8_5	IO		√	—
	P8_6	IO		√	—
	P8_7	IO		√	—
Port group 10	P10_0	IO	General input/output port	√	—
	P10_1	IO		√	—
	P10_2	IO		√	—
	P10_3	IO		√	—
	P10_4	IO		√	—
Port group 11	P11_0	IO	General input/output port	√	√
	P11_1	IO		√	√
	P11_2	IO		√	√
	P11_3	IO		√	√
	P11_4	IO		√	√
	P11_5	IO		√	√
Port group 13	P13_0	IO	General input/output port	√	√
	P13_1	IO		√	√
	P13_2	IO		√	√
	P13_3	IO		√	√
	P13_4	IO		√	√
	P13_5	IO		√	√
	P13_6	IO		√	√
	P13_7	IO		√	√
	P13_8	IO		√	√
	P13_9	IO		√	√
Port group 14	P14_0	I	General input port with edge detection function	√	√
	P14_1	I		√	√
	P14_2	I		√	√
	P14_3	I		√	√
	P14_4	I		√	√
	P14_5	I		√	√
	P14_6	I		√	√
	P14_7	I		√	√
	P14_8	I		√	√
	P14_9	I		√	√
	P14_10	I		√	√

Table 2.72 Pin Function (15/15)

Category	Pin Name	IO	Function	E1M-S BGA304	E1M-S BGA252
Port group 14	P14_11	I	General input port with edge detection function	√	√
	P14_12	I		√	√
	P14_13	I		√	√
	P14_14	I		√	√
	P14_15	I		√	√
Port group 15	P15_0	I	General input port	√	√
	P15_1	I		√	√
	P15_2	I		√	√
	P15_3	I		√	√
Port group 16	P16_0	I	General input port with edge detection function	√	√
	P16_1	I		√	√
	P16_2	I		√	√
	P16_3	I		√	√
	P16_4	I		√	√
	P16_5	I		√	√
	P16_6	I		√	√
	P16_7	I		√	√
	P16_8	I		√	√

2.5.3 Pin State

Definition of Reset State

In description of pin state, each reset state is defined as shown in **Table 2.73**.

Table 2.73 Definition of Reset State

Reset State	Definition
External reset	Reset state from an external pin ($\overline{RES}$ = L)
Internal reset	Between external reset release to internal reset release
After internal reset release	State where internal reset is released

Table 2.74 shows detailed pin states. Some pins may be excluded depending on grades and packages of this product. For the pins included in each product, see **Table 1.1, Pin Numbers and Pin Names (E1MS-BGA304)** and **Table 1.2, Pin Numbers and Pin Names (E1MS-BGA252)** in **Section 1.5, Pin Connection Diagram (Top View)**.

Table 2.74 Pin State (1/2)

Pin Function		Pin State			
		$\overline{RES}$ = L	$\overline{RES}$ = H		
Category	Pin Name (Value after Reset)	External Reset State	Before Internal Reset Release	After Internal Reset Release	Power-off Standby Mode
Clock	CK	O	O	O	Z
	XTAL	O	O	O	Z
	EXTAL	I	I	I	Z
System control	$\overline{RES}$	I (Pull-down)	I (Pull-down)	I (Pull-down)	I (Pull-down)
	MD0	I (Pull-down)	I (Pull-down)	I (Pull-down)	I (Pull-down)
	MD1	I (Pull-down)	I (Pull-down)	I (Pull-down)	I (Pull-down)
	FLMODE	I (Pull-down)	I (Pull-down)	I (Pull-down)	I (Pull-down)
Interrupt	NMI	I (Pull-down)	I (Pull-down)	I (Pull-down)	Z
ECM	$\overline{ERROROUT\_M}$	O	O	O	Z
General input/ output port	P0_0 to P0_14	Z	Z	Z	Z
	P1_0 to P1_15	Z	Z	Z	Z
	P2_0 to P2_7	Z	Z	Z	Z
	P3_0 to P3_7	Z	Z	Z	Z
	P4_0 to P4_15	Z	Z	Z	Z
	P5_0 to P5_9	Z	Z	Z	Z
	P6_0 to P6_11	Z	Z	Z	Z
	P7_0 to P7_11	Z	Z	Z	Z
	P8_0 to P8_7	Z	Z	Z	Z
	P10_0 to P10_4	Z	Z	Z	Z
	P11_0 to P11_5	Z	Z	Z	Z
	P13_0 to P13_9	Z	Z	Z	Z
SAR, ΔΣA/D	AN0xx/DSANxP	Z	Z	Z	Z
	AN0xx/DSANxN	Z	Z	Z	Z
	AN0xx	Z	Z	Z	Z
	AN1xx	Z	Z	Z	Z

Table 2.74 Pin State (2/2)

Pin Function			Pin State			
			$\overline{RES}$ = L	$\overline{RES}$ = H		
Category	Pin Name (Value after Reset)		External Reset State	Before Internal Reset Release	After Internal Reset Release	Power-off Standby Mode
RHSB LVDS buffer	RHSB0FCLP to RHSB1FCLP		Z	Z	Z	Z
	RHSB0FCLN to RHSB1FCLN		Z	Z	Z	Z
	RHSB0SOP to RHSB1SOP		Z	Z	Z	Z
	RHSB0SON to RHSB1SON		Z	Z	Z	Z
AUD RAM monitor	$\overline{AUDRST}$		I (Pull-down)	I (Pull-down)	I (Pull-down)	Z
	AUDCK		I (Pull-up)	I (Pull-up)	I (Pull-up)	Z
	$\overline{AUDSYNC}$		I (Pull-up)	I (Pull-up)	I (Pull-up)	Z
	AUDATA0 to 3		I (Pull-up)	I (Pull-up)	I (Pull-up)	Z
Debug system	$\overline{EVTO}$		O	O	O	Z
	$\overline{EVTI}$		I (Pull-up)	I (Pull-up)	I (Pull-up)	Z
	TDI/LPDI/ FLSCI3RX	Nexus: TDI	Z	I (Pull-up)	I (Pull-up)	Z
		4-pin LPD: LPDI	Z	I (Pull-up)	I (Pull-up)	Z
		Writer I/F: FLSCI3RX	Z	Z	Z	Z
		BSCAN: TDI	I (Pull-up)	I (Pull-up)	I (Pull-up)	Z
	TDO/LPDO/ FLSCI3TX	Nexus: TDO	Z	Z	Z	Z
		4-pin LPD: LPDO	Z	O	O	Z
		Writer I/F: FLSCI3TX	Z	Z	Z	Z
		BSCAN: TDO	Z	Z	Z	Z
	TCK/LPDCLKI/ FLSCI3SCK	Nexus: TCK	Z	I (Pull-up)	I (Pull-up)	Z
		4-pin LPD: LPDCLKI	Z	I (Pull-up)	I (Pull-up)	Z
		Writer I/F: FLSCI3SCK	Z	Z	Z	Z
		BSCAN: TCK	I (Pull-up)	I (Pull-up)	I (Pull-up)	Z
	TMS/$\overline{EVTO}$	Nexus: TMS	Z	I (Pull-up)	I (Pull-up)	Z
		4-pin LPD: $\overline{EVTO}$	Z	O	O	Z
		Writer I/F: (without function)	Z	Z	Z	Z
		BSCAN: TMS	I (Pull-up)	I (Pull-up)	I (Pull-up)	Z
	$\overline{TRST}$/$\overline{LPDRST}$	Nexus: $\overline{TRST}$	I (Pull-down)	I (Pull-down)	I (Pull-down)	I (Pull-down)
		4-pin LPD: $\overline{LPDRST}$				
		Writer I/F: (without function)				
		BSCAN: $\overline{TRST}$				
	$\overline{DRDY}$/LPDCLKO	Nexus: $\overline{DRDY}$	Z	O	O	Z
		4-pin LPD: LPDCLKO	Z	O	O	Z
		Writer I/F: (without function)	Z	Z	Z	Z
		BSCAN: (without function)	Z	Z	Z	Z

Note: I: Input
O: Output
Z: High impedance
Pull-up: On-chip pull-up resistor
Pull-down: On-chip pull-down resistor

2.5.4 Handling of Unused Pins

Examples of the handling of unused pins are listed in **Table 2.75**.

Table 2.75 Examples of the Handling of Unused Pins (1/4)

Classification	Pin	IO	Examples of handling of unused pins	On-chip pull-down/pull-up resistor
Clock	CK	O	• The pin is left open-circuit.	None
	XTAL	O	• The pin is left open-circuit.	None
	EXTAL	I	(This pin will always be used.)	None
System control	$\overline{\text{RES}}$	I	(This pin will always be used.)	This pin is equipped with an internal pull-down resistor.
	MD0, FLMODE	I	(These pins will always be used. When used in user boot mode, it is individually connected to VSS via a resistor.)	This pin is equipped with an internal pull-down resistor.
	MD1	I	• This pin is individually connected to VSS via a resistor.	This pin is equipped with an internal pull-down resistor.
Interrupt	NMI	I	• This pin is individually connected to VSS via a resistor.	This pin is equipped with an internal pull-down resistor.
ECM	$\overline{\text{ERROROUT\_M}}$	O	• The pin is left open-circuit.	None
General input/ output port	P0_m, P1_m, P2_m, P3_m, P4_m, P5_m, P6_m, P7_m, P8_m, P10_m, P11_m, P13_m	IO	[Input mode] • These pins are left open-circuit and the settings are to disable input (values after reset are PMCn_m = 0, PMn_m = 1, PIBCn_m = 0) • These pins are left open-circuit and the settings are to enable on-chip pull-up/pull-down resistors (use PUn_m,PDn_m) • These pins are individually connected to power supply or ground of each pin via a resistor. [Output mode] • These pins are left open-circuit.	The pins are equipped with internal pull-up or pull-down resistors which can be enabled by register settings.
ADCB, DS-ADC	AN000/DSAN0P, AN001/DSAN0N, AN002/DSAN1P, AN003/DSAN1N, AN10x, AN11x, AN12x, AN13x, AN14x (Analog input-only)	I	• These pins are left open-circuit.	None
	AN010/DSAN2P, AN011/DSAN2N, AN012/DSAN3P, AN013/DSAN3N, AN020/DSAN4P, AN021/DSAN4N, AN022/DSAN5P, AN023/DSAN5N, AN030/DSAN6P, AN031/DSAN6N, AN032/DSAN7P, AN033/DSAN7N, AN04x, AN05x, AN15x, AN16x, AN17x (Alternative analog input and digital input)	I	• These pins are left open-circuit and the settings are to disable input (values after reset are PMCn_m = 0, and PIBCn_m = 0)	None

Table 2.75 Examples of the Handling of Unused Pins (2/4)

Classification	Pin		IO	Examples of handling of unused pins	On-chip pull-down/pull-up resistor
RHSB LVDS buffer	RHSBxSOP, RHSBxSON, RHSBxFCLP, RHSBxFCLN		O	• These pins are left open-circuit and the settings are to disable output (value after reset is PM17_m = 1).	None
AUD RAM monitor	$\overline{\text{AUDRST}}$		I	• The pin is left open-circuit. • This pin is individually connected to VSS via a resistor.	This pin is equipped with an internal pull-down resistor.
	AUDCK, $\overline{\text{AUDSYNC}}$		I	• These pins are left open-circuit. • These pins are individually connected to VCC via a resistor.	This pin is equipped with an internal pull-up resistor.
	AUDATAn		IO	• The pin is left open-circuit.	This pin is equipped with an internal pull-up resistor.
Debug	$\overline{\text{EVTO}}$		O	• The pin is left open-circuit.	None
	$\overline{\text{EVTI}}$		I	• The pin is left open-circuit. • This pin is individually connected to VCC via a resistor.	This pin is equipped with an internal pull-up resistor.
	TDI/LPDI/ FLSCI3RX	TDI	I	• The pin is left open-circuit. • This pin is individually connected to VCC via a resistor.	This pin is equipped with an internal pull-up resistor.
		LPDI	I	• The pin is left open-circuit. • This pin is individually connected to VCC via a resistor.	This pin is equipped with an internal pull-up resistor.
		FLSCI3RX	I	• Do not select the FLSCI3RX pin function if the pin is not use in use.	The internal pull-up resistor is enabled in booting up by the boot firmware.
	TDO/LPDO/ FLSCI3TX	TDO	O	• The pin is left open-circuit.	None
		LPDO	O	• The pin is left open-circuit.	None
		FLSCI3TX	O	• Do not select the FLSCI3TX pin function if the pin is not use in use.	None
	TCK/ LPDCLKI/ FLSCI3SCK	TCK	I	• The pin is left open-circuit. • This pin is individually connected to VCC via a resistor.	This pin is equipped with an internal pull-up resistor.
		LPDCLKI	I	• The pin is left open-circuit. • This pin is individually connected to VCC via a resistor.	This pin is equipped with an internal pull-up resistor.
		FLSCI3SCK	I	• Do not select the FLSCI3SCK pin function if the pin is not use in use.	None
	TMS/$\overline{\text{EVTO}}$	TMS	I	• The pin is left open-circuit. • This pin is individually connected to VCC via a resistor.	This pin is equipped with an internal pull-up resistor.
		$\overline{\text{EVTO}}$	O	• The pin is left open-circuit.	None
	$\overline{\text{TRST}}$/$\overline{\text{LPDRST}}$		I	• This pin is individually connected to VSS via a resistor.	This pin is equipped with an internal pull-down resistor.
	$\overline{\text{DRDY}}$/ LPDCLKO	$\overline{\text{DRDY}}$	O	• The pin is left open-circuit.	None
		LPDCLKO	O	• The pin is left open-circuit.	None
NC	NC		—	• The pin is left open-circuit.	None

Table 2.75 Examples of the Handling of Unused Pins (3/4)

Classification	Pin	IO	Examples of handling of unused pins	On-chip pull-down/pull-up resistor
Power supply	EVCC		(This pin will always be used.)	—
	SYSVCC		(This pin will always be used.)	—
	VCC		(This pin will always be used.)	—
	PLLVCC		(This pin will always be used.)	—
	PLLVSS		(This pin will always be used.)	—
	VDD		(This pin will always be used.)	—
	RAMVCL		(This pin will always be used.)	—
	VSS		(This pin will always be used.)	—
	LVDVCC		• This pin is connected to VCC.	—
	LVDVSS		• This pin is connected to VSS.	—
	TTLVCC		• This pin is connected to VCC.	—
	EPTVOUT		• The pin is left open-circuit.	—
	ADSVCC, ADSVSS, ADSVREFH, ADSVREFL, ADSVCL		• Connections when the DS-ADC is not in use but the ADCB module is are listed below. ADSVCC : This pin is connected to A0VCC. ADSVSS : This pin is connected to A0VSS. ADSVREFH: This pin is connected to A0VCC. ADSVREFL: This pin is connected to A0VSS. ADSVCL : The pin is left open-circuit. • Connections when neither the DS-ADC nor the ADCB module is in use are listed below. ADSVCC : This pin is connected to EVCC. ADSVSS : This pin is connected to VSS. ADSVREFH: This pin is connected to EVCC. ADSVREFL: This pin is connected to VSS. ADSVCL : The pin is left open-circuit.	—

Table 2.75 Examples of the Handling of Unused Pins (4/4)

Classification	Pin	IO	Examples of handling of unused pins	On-chip pull-down/pull-up resistor
Power supply	A0VCC, A0VSS, A0VREFH, A1VCC, A1VSS, A1VREFH		• Connections when the ADCB module is not in use but the ADC is are listed below. A0VCC : This pin is connected to ADSVCC. A0VSS : This pin is connected to ADSVSS. A0VREFH : This pin is connected to ADSVCC. A1VCC : This pin is connected to ADSVCC. A1VSS : This pin is connected to ADSVSS. A1VREFH : This pin is connected to ADSVCC. • Connections when neither the DS-ADC nor the ADCB module is in use are listed below. A0VCC : This pin is connected to EVCC. A0VSS : This pin is connected to VSS. A0VREFH : This pin is connected to EVCC. A1VCC : This pin is connected to EVCC. A1VSS : This pin is connected to VSS. A1VREFH : This pin is connected to EVCC.	—

Note 1. For the correspondences between the various pins and the power supply or ground, see **Section 37 Electrical Characteristics**.

Note 2. Use a resistor with a value of at least 1 kΩ for externally pulling pins up or down.

Note 3. Take care regarding the resistive division effect when using an external resistor to pull up or pull down a pin that is equipped with an internal pull-up or pull-down resistor.

Section 3 CPU System

3.1 Overview

3.1.1 Block Configuration

Figure 3.1 shows the block configuration diagram.

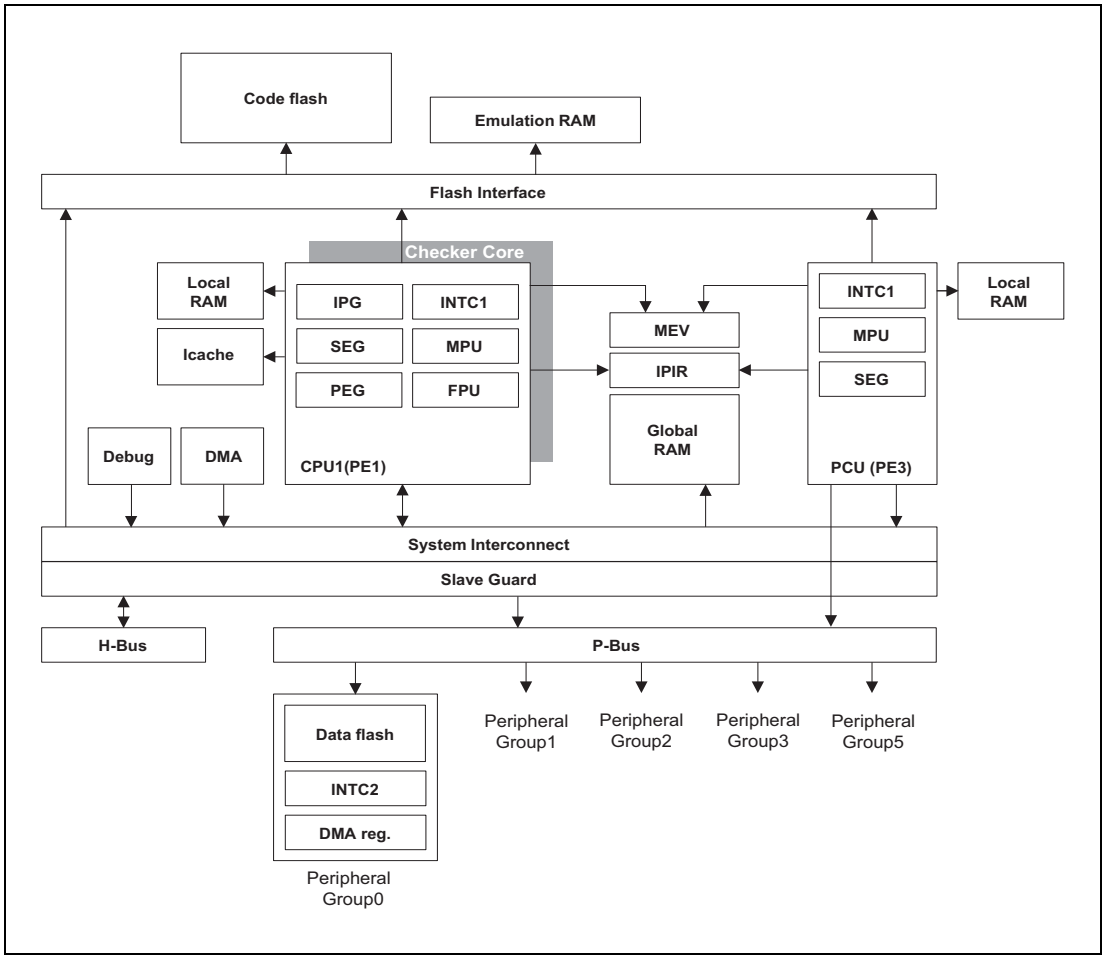


Figure 3.1 Block Configuration Diagram

CPU1 (PE1)

The RH850 G3M core is included as a main CPU. CPU1 also includes the checker core for safety assurance.

PCU (PE3)

The RH850 G3K core is included as a PCU to control peripheral IPs (peripheral control unit).

Local RAM

Each PE has a high-speed accessible local RAM.

Global RAM

The global RAM is a mass on-chip RAM that all PEs share. Some areas of this RAM function as the retention RAM.

Code Flash

A mass code flash is included for program storage. CPU1 and PCU share the code flash and they are connected with each other via the code flash interface.

Emulation RAM

It is a RAM to emulate the code flash. The programs can be replaced by controlling from an external tool without rewriting the code flash.

Data Flash

It is a flash memory being rewritable by the CPU.

P-Bus and H-Bus

The P-Bus and H-Bus connect the peripheral IPs. The P-Bus is divided into five, Peripheral Group 0 to 3, and 5.

For the correspondence between the peripheral groups and the peripheral modules/registers, see **Appendix, List of Registers**.

INTC1, INTC2

INTC1 is an interrupt controller exclusive to each PE. INTC2 is a common interrupt controller that all PEs share, being able to set the binding destination PE of an interrupt request by the registers.

DMA

Two DMA transfer modules, DMAC and DTS, are included.

Slave Guard

The slave guard is a system to prevent unauthorized access from the specific bus master, consisting of the following guard structures:

(1) PE Guard (PEG)

The PE guard is a system to prevent unauthorized access to the resources in the PE from the external master. Access from the PE itself is only enabled but all other accesses are disabled after release from the reset state.

(2) Global RAM Guard (GRG)

The global RAM guard is a system to prevent unauthorized access to the global RAM from the external master. The global RAM is in the unprotected state (accessible from all bus masters) after release from the reset state.

(3) Peripheral Guard (PBG)

The peripheral guard is a system to prevent unauthorized access to peripheral devices from the external master. Peripheral devices are in the unprotected state (accessible from all bus masters) after release from the reset state.

3.2 CPU

3.2.1 Core Functions

3.2.1.1 Features

Table 3.1 lists features of the RH850G3M core.

Table 3.1 Features of the RH850G3M Core

Item	Feature
CPU	• Advanced 32-bit architecture for embedded control • 32-bit internal data bus • Thirty-two 32-bit general-purpose registers • RISC-type instruction sets – Long-/short-format load/store instructions – Three-operand instructions – Instruction sets based on C language • CPU operating modes – User mode, supervisor mode • Address space: 4-Gbyte linear address space for both data and instructions • Instructions: A snooze instruction (SNOOZE) is included for temporary suspension by switching the CPU clock signal (CLK_CPU) off for 32 clock cycles.
Coprocessor	• A floating-point operation coprocessor (FPU) mounted – Supports single precision (32 bits) and double precision (64 bits). – Supports data types and exceptions conforming to IEEE754. – Rounding mode: Neighborhood, 0 direction, +∞ direction, and -∞ direction – Handling denormalized numbers: Rounding down to 0 or exception notification to conform to IEEE754
Exception/ Interrupt	• 16 interrupt priority levels settable for each channel • Vector selection method selectable according to performance request or memory usage – Direct branching exception vectors – Indirect branching exception vectors referring to the address table • Supports the high-speed save/restore processing of the context by the dedicated instructions (PUSHSP and POPSP) at the generation of an interrupt
Memory Management	• Memory protection function (MPU): 12 areas settable
Cache	• Instruction cache

3.2.1.2 Register Set

This subsection explains the program registers and system registers incorporated in this CPU.

(1) Program Registers

Program registers include the general-purpose registers (r0 to r31) and program counter (PC).

The general-purpose register r0 always retains 0, but the values of the registers r1 to r31 after reset are undefined.

Table 3.2 List of Program Registers

Program Register	Name	Function	Description
General-purpose registers	r0	Zero register	Always retains “0”
	r1	Assembler reserved register	Used as working register for generating addresses
	r2	Register for address and data variables (used when the real-time OS used does not use this register)	
	r3	Stack pointer (SP)	Used for generating a stack frame when a function is called
	r4	Global pointer (GP)	Used for accessing a global variable in the data area
	r5	Text pointer (TP)	Used as a register that indicates the start of the text area (area where program code is placed)
	r6 to r29	Register for addresses and data variables	
	r30	Element pointer (EP)	Used as a base pointer for generating addresses when accessing memory
	r31	Link pointer (LP)	Used when the compiler calls a function
Program counter	PC	Retains instruction addresses during execution of programs	

Note: For further descriptions of r1, r3 to r5, and r31 used for an assembler and/or C compiler, see the specification of each software development environment.

(a) General-Purpose Registers

A total of 32 general-purpose registers (r0 to r31) are provided. All of these registers can be used for either data variables or address variables.

Of the general-purpose registers, r0 to r5, r30, and r31 are assumed to be used for special purposes in software development environments, so it is necessary to note the following when using them.

1. r0, r3, and r30

   These registers are implicitly used by instructions.

   r0 is a register that always retains "0". It is used for operations that use 0 and addressing with base address being 0.

   r3 is implicitly used by the PREPARE, DISPOSE, PUSHSP, and POPSP instructions.

   r30 is used as a base pointer when the SLD instruction or SST instruction accesses memory.

2. r1, r4, r5, and r31

   These registers are implicitly used by the assembler and C compiler.

   When using these registers, register contents must first be saved so they are not lost and can be restored after the registers are used.

3. r2

   This register might be used by a real-time OS in some cases. If the real-time OS that is being used is not using r2, r2 can be used as a register for address variables or data variables.

(b) PC — Program Counter

The PC retains the address of the instruction being executed.

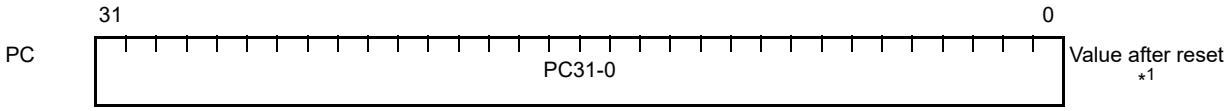


Table 3.3 PC Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 1	PC31-1	These bits indicate the address of the instruction being executed.	R/W	*1
0	PC0	This bit is fixed to 0, and branching to an odd number address is disabled.	R/W	0

Note 1. The value after reset depends on the startup area in this product. For details, see **Section 4, Address Space**

(2) Basic System Registers

The basic system registers are used to control CPU status and to retain exception information.

System registers are read from or written to by using the LDSR or STSR instructions and specifying the system register number, which is made up of a register number and selection ID.

Table 3.4 Basic System Registers

Register No. (reg.ID, sel.ID)	Symbol	Function	Access Permission
SR0, 0	EIPC	Status save registers when acknowledging EI level exception	SV
SR1, 0	EIPSW	Status save registers when acknowledging EI level exception	SV
SR2, 0	FEPC	Status save registers when acknowledging FE level exception	SV
SR3, 0	FEPSW	Status save registers when acknowledging FE level exception	SV
SR 5, 0	PSW	Program status word	*1
SR 6, 0	FPSR	Refer to **(4) FPU Function Registers**	CU0 and SV
SR 7, 0	FPEPC	Refer to **(4) FPU Function Registers**	CU0 and SV
SR8, 0	FPST	Refer to **(4) FPU Function Registers**	CU0
SR9, 0	FPCC	Refer to **(4) FPU Function Registers**	CU0
SR10, 0	FPCFG	Refer to **(4) FPU Function Registers**	CU0
SR 11, 0	FPEC	Refer to **(4) FPU Function Registers**	CU0 and SV
SR 13, 0	EIIC	EI level exception cause	SV
SR 14, 0	FEIC	FE level exception cause	SV
SR 16, 0	CTPC	CALLT execution status save register	UM
SR 17, 0	CTPSW	CALLT execution status save register	UM
SR 20, 0	CTBP	CALLT base pointer	UM
SR 28, 0	EIWR	EI level exception working register	SV
SR 29, 0	FEWR	FE level exception working register	SV
SR 31, 0	(BSEL)	(Reserved for backwards compatibility with the V850E2 series)*2	SV
SR0, 1	MCFG0	Machine configuration	SV
SR2, 1	RBASE	Reset vector base address	SV
SR3, 1	EBASE	Exception handler vector address	SV
SR4, 1	INTBP	Base address of the interrupt handler table	SV
SR 5, 1	MCTL	CPU control	SV
SR 6, 1	PID	Processor ID	SV
SR 11, 1	SCCFG	SYSCALL operation setting	SV
SR 12, 1	SCBP	SYSCALL base pointer	SV
SR0, 2	HTCFG0	Thread configuration	SV
SR6, 2	MEA	Memory error address	SV
SR7, 2	ASID	Address space ID	SV
SR8, 2	MEI	Memory error information	SV

Note 1. The access permission differs depending on the bit. For details, see **(e) PSW — Program status word** in **(2) Basic System Registers**.

Note 2. Reserved for backwards compatibility with the V850E2 series. These registers are always read as 0. Writing to these registers is ignored.

(a) EIPC — Status save register when acknowledging EI level exception

When an EI level exception is acknowledged, the address of the instruction that was being executed when the EI level exception occurred, or of the next instruction, is saved to the EIPC register (see *section 4.1.3 Types of exceptions* in *RH850G3M User's Manual: Software*).

Because there is only one pair of EI level exception status save registers, when processing multiple exceptions, the contents of these registers must be saved by a program.

Be sure to set an even-numbered address to the EIPC register. An odd-numbered address must not be specified.

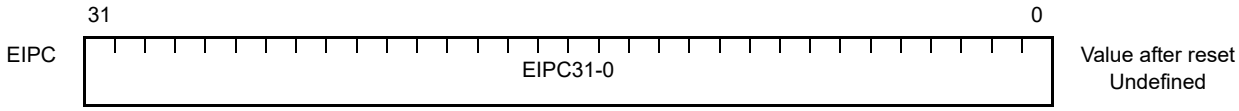


Table 3.5 EIPC Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 1	EIPC31-1	These bits indicate the PC saved when an EI level exception is acknowledged.	R/W	Undefined
0	EIPC0	This bit indicates the PC saved when an EI level exception is acknowledged. Always set this bit to 0. Even if it is set to 1, the value transferred to the PC when the EIRET instruction is executed is 0.	R/W	Undefined

(b) EIPSW — Status save register when acknowledging EI level exception

When an EI level exception is acknowledged, the current PSW setting is saved to the EIPSW register.

Because there is only one pair of EI level exception status save registers, when processing multiple exceptions, the contents of these registers must be saved by a program.

CAUTION

Bits 11 to 9 cannot usually be changed since these bits are associated with debugging.

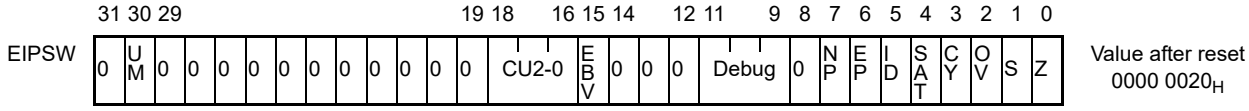


Table 3.6 EIPSW Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
30	UM	This bit stores the PSW.UM bit setting when an EI level exception is acknowledged.	R/W	0
29 to 19	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
18 to 16	CU2-0	These bits store the PSW.CU2 to 0 field setting when an EI level exception is acknowledged. (CU2 and CU1 are reserved for future expansion. Be sure to clear to 0.)	R/W	0
15	EBV	This bit stores the PSW.EBV bit setting when an EI level exception is acknowledged.	R/W	0
14 to 12	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
11 to 9	Debug	These bits store the PSW.Debug field when an EI level exception is acknowledged.	R/W	0
8	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
7	NP	This bit stores the PSW.NP bit setting when an EI level exception is acknowledged.	R/W	0
6	EP	This bit stores the PSW.EP bit setting when an EI level exception is acknowledged.	R/W	0
5	ID	This bit stores the PSW.ID bit setting when an EI level exception is acknowledged.	R/W	1
4	SAT	This bit stores the PSW.SAT bit setting when an EI level exception is acknowledged.	R/W	0
3	CY	This bit stores the PSW.CY bit setting when an EI level exception is acknowledged.	R/W	0
2	OV	This bit stores the PSW.OV bit setting when an EI level exception is acknowledged.	R/W	0
1	S	This bit stores the PSW.S bit setting when an EI level exception is acknowledged.	R/W	0
0	Z	This bit stores the PSW.Z bit setting when an EI level exception is acknowledged.	R/W	0

(c) FEPC — Status save register when acknowledging FE level exception

When an FE level exception is acknowledged, the address of the instruction that was being executed when the FE level exception occurred, or of the next instruction, is saved to the FEPC register (see *section 4.1.3 Types of exceptions* in *RH850G3M User's Manual: Software*).

Because there is only one pair of FE level exception status save registers, when processing multiple exceptions, the contents of these registers must be saved by a program.

Be sure to set an even-numbered address to the FEPC register. An odd-numbered address must not be specified.

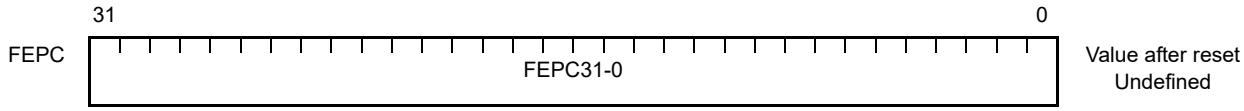


Table 3.7 FEPC Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 1	FEPC31-1	These bits indicate the PC saved when an FE level exception is acknowledged.	R/W	Undefined
0	FEPC0	This bit indicates the PC saved when an FE level exception is acknowledged. Always set this bit to 0. Even if it is set to 1, the value transferred to the PC when the FERET instruction is executed is 0.	R/W	Undefined

(d) FEPSW — Status save register when acknowledging FE level exception

When an FE level exception is acknowledged, the current PSW setting is saved to the FEPSW register.

Because there is only one pair of FE level exception status save registers, when processing multiple exceptions, the contents of these registers must be saved by a program.

CAUTION

Bits 11 to 9 cannot usually be changed since these bits are associated with debugging.

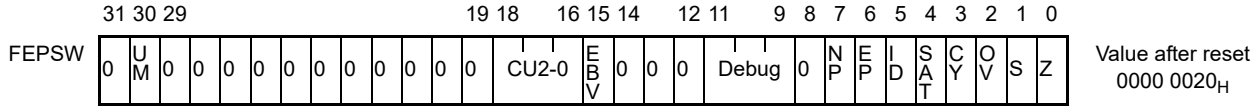


Table 3.8 FEPSW Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31	—	(Reserved for future expansion. Be sure to clear to 0.)	R/W	0
30	UM	This bit stores the PSW.UM bit setting when an FE level exception is acknowledged.	R/W	0
29 to 19	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
18 to 16	CU2-0	These bits store the PSW.CU2 to 0 field setting when an FE level exception is acknowledged. (CU2 and CU1 are reserved for future expansion. Be sure to clear to 0.)	R/W	0
15	EBV	This bit stores the PSW.EBV bit setting when an FE level exception is acknowledged.	R/W	0
14 to 12	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
11 to 9	Debug	These bits store the PSW.Debug field when an EI level exception is acknowledged.	R/W	0
8	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
7	NP	This bit stores the PSW.NP bit setting when an FE level exception is acknowledged.	R/W	0
6	EP	This bit stores the PSW.EP bit setting when an FE level exception is acknowledged.	R/W	0
5	ID	This bit stores the PSW.ID bit setting when an FE level exception is acknowledged.	R/W	1
4	SAT	This bit stores the PSW.SAT bit setting when an FE level exception is acknowledged.	R/W	0
3	CY	This bit stores the PSW.CY bit setting when an FE level exception is acknowledged.	R/W	0
2	OV	This bit stores the PSW.OV bit setting when an FE level exception is acknowledged.	R/W	0
1	S	This bit stores the PSW.S bit setting when an FE level exception is acknowledged.	R/W	0
0	Z	This bit stores the PSW.Z bit setting when an FE level exception is acknowledged.	R/W	0

(e) PSW — Program status word

PSW (program status word) is a set of flags that indicate the program status (instruction execution result) and bits that indicate the operation status of the CPU (flags are bits in the PSW that are referenced by a condition instruction (Bcond, CMOV, etc.)).

CAUTIONS

1. **When the LDSR instruction is used to change the contents of bits 7 to 0 in this register, the changed contents become valid immediately after the completion of execution of the LDSR instruction.**
2. **The access permission for the PSW register differs with each bit. All bits can be read, but some bits can only be written under certain conditions. See Table 3.9, Access Permission for PSW Register for the access permission for each bit.**

Table 3.9 Access Permission for PSW Register

Bit		Access Permission When Reading	Access Permission When Writing
30	UM	UM	SV*1
18 to 16	CU2-0		SV*1
15	EBV		SV*1
11 to 9	Debug		Special*1
7	NP		SV*1
6	EP		SV*1
5	ID		SV*1
4	SAT		UM
3	CY		UM
2	OV		UM
1	S		UM
0	Z		UM

Note 1. The access permission for the whole PSW register is UM, so the PIE exception does not occur even if the register is written by using an LDSR instruction when PSW.UM is 1. In this case, writing is ignored.

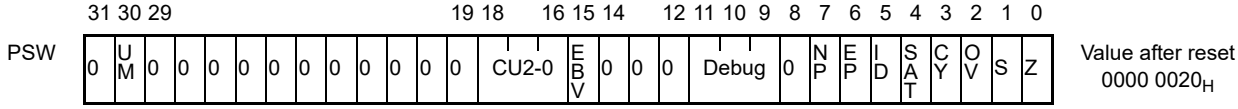


Table 3.10 PSW Register Contents (1/2)

Bit Position	Bit Name	Function	R/W	Value after Reset
31	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
30	UM	This bit indicates that the CPU is in user mode (UM mode) 0: Supervisor mode 1: User mode	R/W	0
29 to 19	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
18 to 16	CU2-CU0	This bit indicates the coprocessor use permissions. When the bit corresponding to the coprocessor is 0, a coprocessor use prohibition exception is generated in response to execution of a coprocessor instruction or access to coprocessor resources (system registers). CU2 bit 18: (Reserved for future expansion. Be sure to clear to 0.) CU1 bit 17: (Reserved for future expansion. Be sure to clear to 0.) CU0 bit 16: FPU	R/W	000

Table 3.10 PSW Register Contents (2/2)

Bit Position	Bit Name	Function	R/W	Value after Reset
15	EBV	This bit indicates the reset vector and exception vector operation. See the descriptions of the RBASE register in **(q) RBASE — Reset vector base address register** and the EBASE register in **(r) EBASE — Exception handler vector address register**.	R/W	0
14 to 12	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
11 to 9	Debug	These bits are used in debugging for development tools. In normal operation, clear these bits to 0.	—	0
8	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
7	NP	Disables acceptance of FE level exceptions. Once an FE level exception is accepted, this bit is set to 1 and disables acceptance of EI level exceptions or FE level exceptions. For the exceptions disabled by the NP bit, see *table 4-1 Exception Cause List* in *RH850G3M User's Manual: Software*. 0 : An acceptance of FE level exceptions is enabled 1 : An acceptance of FE level exceptions is disabled	R/W	0
6	EP	This bit indicates that an exception other than an interrupt is being serviced. It is set to 1 when the corresponding exception occurs. This bit does not affect acknowledging an exception request even when it is set to 1. 0: An exception other than an interrupt is not being serviced. 1: An exception other than an interrupt is being serviced.	R/W	0
5	ID	Disables acceptance of EI level exceptions. Once an IE level exception or FE level exception is accepted, this bit is set to 1 and disables acceptance of EI level exceptions. For the exceptions disabled by the ID bit, see *table 4-1 Exception Cause List* in *RH850G3M User's Manual: Software*. This bit is also used to disable EI level exceptions from being acknowledged as a critical section while an ordinary program or interrupt is being serviced. It is set to 1 when the DI instruction is executed, and cleared to 0 when the EI instruction is executed. A change in the ID bit by the IE or DI instruction is effective from the next instruction. 0 : An acceptance of IE level exceptions is enabled 1 : An acceptance of IE level exceptions is disabled	R/W	1
4	SAT*[1]	This bit indicates that the operation result is saturated because the result of a saturated operation instruction operation has overflowed. This is a cumulative flag, so when the operation result of the saturated operation instruction becomes saturated, this bit is set to 1, but it is not cleared to 0 when the operation result for a subsequent instruction is not saturated. This bit is cleared to 0 by the LDSR instruction. This bit is neither set to 1 nor cleared to 0 when an arithmetic operation instruction is executed. 0: Not saturated 1: Saturated	R/W	0
3	CY	This bit indicates whether a carry or borrow has occurred in the operation result. 0: Carry and borrow have not occurred. 1: Carry or borrow has occurred.	R/W	0
2	OV*[1]	This bit indicates whether or not an overflow has occurred during an operation. 0: Overflow has not occurred. 1: Overflow has occurred.	R/W	0
1	S*[1]	This bit indicates whether or not the result of an operation is negative. 0: Result of operation is positive or 0. 1: Result of operation is negative	R/W	0
0	Z	This bit indicates whether or not the result of an operation is 0. 0: Result of operation is not 0. 1: Result of operation is 0.	R/W	0

Note 1. The operation result of the saturation processing is determined in accordance with the contents of the OV flag and S flag during a saturated operation. When only the OV flag is set to 1 during a saturated operation, the SAT flag is set to 1.

Operation result status	Flag status			Operation result after saturation processing
	SAT	OV	S	
Exceeded positive maximum value	1	1	0	7FFF FFFF$_H$
Exceeded negative maximum value	1	1	1	8000 0000$_H$
Positive (maximum value not exceeded)	Value prior to operation is retained.	0	0	Operation result itself
Negative (maximum value not exceeded)			1	

(f) EIIC — EI level exception source register

The EIIC register retains the source of any EI level exception that occurs. The value retained in this register is an exception code corresponding to a specific exception source.

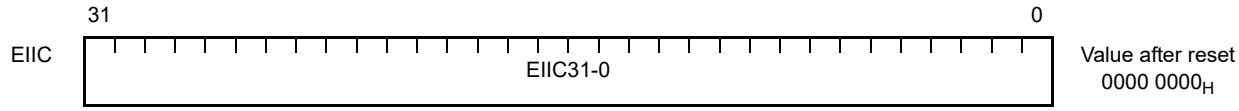


Table 3.11 EIIC Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 0	EIIC31-0	These bits store the exception source code when an EI level exception occurs. Regarding the stored exception source code, see *Table 6.14, Interrupt Exception Handler and Priority* and *table 4-1 Exception Cause List* in *RH850G3M User's Manual: Software*. The EIIC31-16 field stores detailed exception source codes defined individually for each exception. If there is no particular definition, these bits are set to 0.	R/W	0

(g) FEIC — FE level exception source register

The FEIC register retains the source of any FE level exception that occurs. The value retained in this register is an exception code corresponding to a specific exception source.

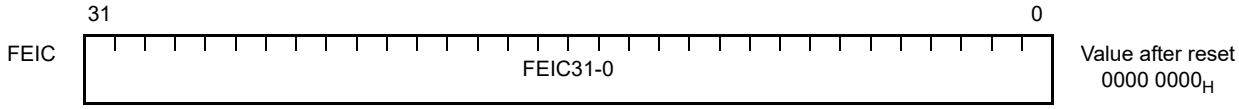


Table 3.12 FEIC Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 0	FEIC31-0	These bits store the exception source code when an FE level exception occurs. Regarding the stored exception source code, see *Table 6.14, Interrupt Exception Handler and Priority* and *table 4-1 Exception Cause List* in *RH850G3M User's Manual: Software*. The FEIC31-16 field stores detailed exception source codes defined individually for each exception. If there is no particular definition, these bits are set to 0.	R/W	0

(h) CTPC — Status save register when executing CALLT register

When a CALLT instruction is executed, the address of the next instruction after the CALLT instruction is saved to CTPC. Be sure to set an even-numbered address to the CTPC register. An odd-numbered address must not be specified.

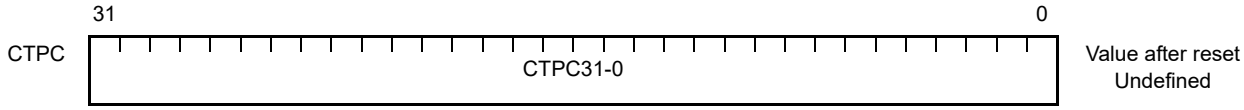


Table 3.13 CTPC Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 1	CTPC31-1	These bits indicate the PC of the instruction after the CALLT instruction.	R/W	Undefined
0	CTPC0	This bit indicates the PC of the instruction after the CALLT instruction. Always set this bit to 0. Even if it is set to 1, the value transferred to the PC when the CTRET instruction is executed is 0.	R/W	Undefined

(i) CTPSW — Status save register when executing CALLT register

When a CALLT instruction is executed, some of the PSW (program status word) settings are saved to CTPSW.

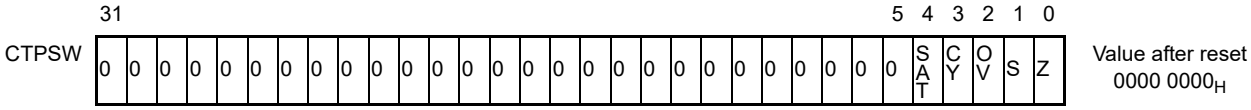


Table 3.14 CTPSW Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 5	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
4	SAT	This bit stores the PSW.SAT bit setting when the CALLT instruction is executed.	R/W	0
3	CY	This bit stores the PSW.CY bit setting when the CALLT instruction is executed.	R/W	0
2	OV	This bit stores the PSW.OV bit setting when the CALLT instruction is executed.	R/W	0
1	S	This bit stores the PSW.S bit setting when the CALLT instruction is executed.	R/W	0
0	Z	This bit stores the PSW.Z bit setting when the CALLT instruction is executed.	R/W	0

(j) CTBP — CALLT base pointer register

The CTBP register is used to specify table addresses of the CALLT instruction and generate target addresses. Be sure to set the CTBP register to a half word address.

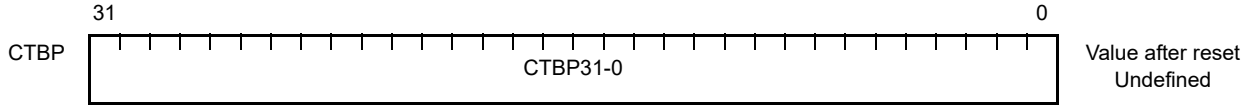


Table 3.15 CTBP Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 1	CTBP31-1	These bits indicate the base pointer address of the CALLT instruction. These bits indicate the first address in the table used by the CALLT instruction.	R/W	Undefined
0	CTBP0	These bits indicate the base pointer address of the CALLT instruction. These bits indicate the first address in the table used by the CALLT instruction. Always set this bit to 0.	R	0

(k) ASID — Address space ID register

This is the address space ID. This is used to identify the address space provided by the memory management function.

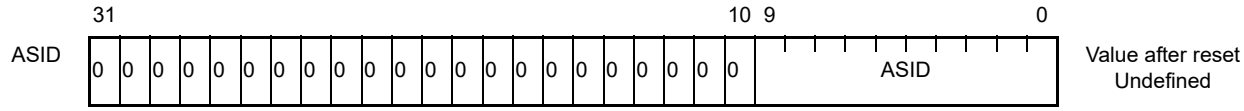


Table 3.16 ASID Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 10	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
9 to 0	ASID	This is the address space ID.	R/W	Undefined

(l) EIWR — EI level exception working register

The EIWR register is used as a working register when an EI level exception has occurred.

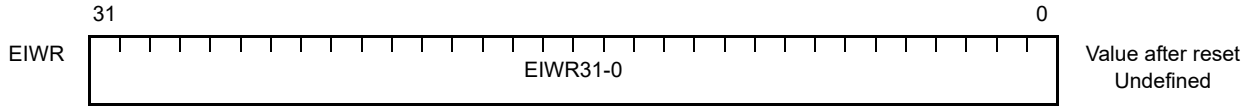


Table 3.17 EIWR Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 0	EIWR31-0	These bits constitute a working register that can be used for any purpose during the processing of an EI level exception. Use this register for purposes such as storing the values of general-purpose registers.	R/W	Undefined

(m) FEWR — FE level exception working register

The FEWR register is used as a working register when an FE level exception has occurred.

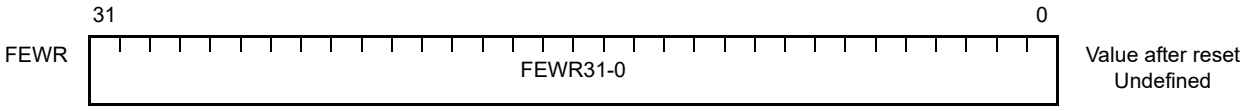


Table 3.18 FEWR Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 0	FEWR31-0	These bits constitute a working register that can be used for any purpose during the processing of an FE level exception. Use this register for purposes such as storing the values of general-purpose registers.	R/W	Undefined

(n) HTCFG0 — Thread configuration register

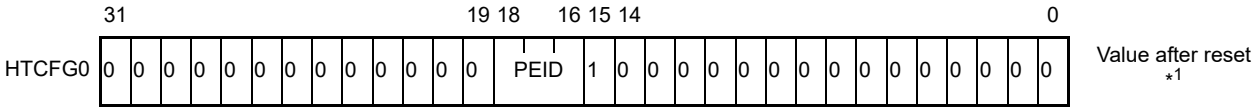


Table 3.19 HTCFG0 Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 19	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
18 to 16	PEID	These bits indicate the processor element number.	R	*2
15	—	(Reserved for future expansion. Be sure to set this bit to 1.)	R	1
14 to 0	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0

Note 1. The value is 0001 8000_H in CPU1 (PE1) of this product.
Note 2. The value is 001_B in CPU1 (PE1) of this product.

(o) MEA — Memory error address register

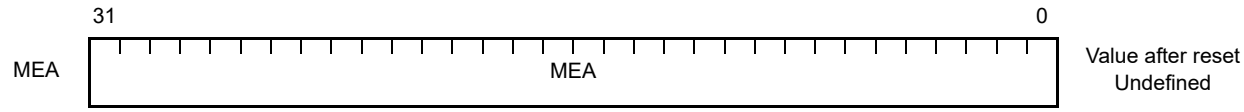


Table 3.20 MEA Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 0	MEA	These bits store an address when a MAE (misaligning) or MPU violation occurs.	R/W	Undefined

(p) MEI — Memory error information register

This register is used to store information about the instruction that caused the exception when a misaligned (MAE) or memory protection (MDP) exception occurs.

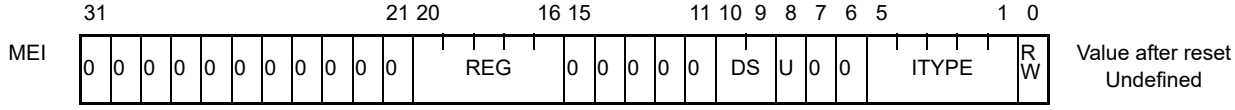


Table 3.21 MEI Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 21	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
20 to 16	REG4-0	These bits indicate the number of the source or destination register accessed by the instruction that caused the exception. For details, see **Table 3.22, Instructions Causing Exceptions and Values of MEI Register**.	R/W	Undefined
15 to 11	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
10, 9	DS	These bits indicate the type of data handled by the instruction that caused the exception.*1 0: Byte (8 bits) 1: Half word (16 bits) 2: Word (32 bits) 3: Double-word (64 bits) For details, see **Table 3.22, Instructions Causing Exceptions and Values of MEI Register**.	R/W	Undefined
8	U	These bits indicate the sign extension method of the instruction that caused the exception. 0: Signed 1: Unsigned For details, see **Table 3.22, Instructions Causing Exceptions and Values of MEI Register**.	R/W	Undefined
7, 6	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
5 to 1	ITYPE4-0	These bits indicate the instruction that caused the exception. For details, see **Table 3.22, Instructions Causing Exceptions and Values of MEI Register**.	R/W	Undefined
0	RW	This bit indicates whether the operation of the instruction that caused the exception was read (Load-memory) or write (Store-memory) 0: Read (Load-memory) 1: Write (Store-memory) For details, see **Table 3.22, Instructions Causing Exceptions and Values of MEI Register**.	R/W	Undefined

Note 1. Even if the data is divided and access is made several times due to the specifications of the hardware, the original data type indicated by the instruction is stored.

Table 3.22 Instructions Causing Exceptions and Values of MEI Register (1/2)

Instruction	REG	DS	U	RW	ITYPE
SLD.B	dst	0 (Byte)	0 (Signed)	0 (Read)	00000_B
SLD.BU	dst	0 (Byte)	1 (Unsigned)	0 (Read)	00000_B
SLD.H	dst	1 (Half-word)	0 (Signed)	0 (Read)	00000_B
SLD.HU	dst	1 (Half-word)	1 (Unsigned)	0 (Read)	00000_B
SLD.W	dst	2 (Word)	0 (Signed)	0 (Read)	00000_B
SST.B	src	0 (Byte)	0 (Signed)	1 (Write)	00000_B
SST.H	src	1 (Half-word)	0 (Signed)	1 (Write)	00000_B
SST.W	src	2 (Word)	0 (Signed)	1 (Write)	00000_B
LD.B (disp16)	dst	0 (Byte)	0 (Signed)	0 (Read)	00001_B

Table 3.22 Instructions Causing Exceptions and Values of MEI Register (2/2)

Instruction	REG	DS	U	RW	ITYPE
LD.BU (disp16)	dst	0 (Byte)	1 (Unsigned)	0 (Read)	00001_B
LD.H (disp16)	dst	1 (Half-word)	0 (Signed)	0 (Read)	00001_B
LD.HU (disp16)	dst	1 (Half-word)	1 (Unsigned)	0 (Read)	00001_B
LD.W (disp16)	dst	2 (Word)	0 (Signed)	0 (Read)	00001_B
ST.B (disp16)	src	0 (Byte)	0 (Signed)	1 (Write)	00001_B
ST.H (disp16)	src	1 (Half-word)	0 (Signed)	1 (Write)	00001_B
ST.W (disp16)	src	2 (Word)	0 (Signed)	1 (Write)	00001_B
LD.B (disp23)	dst	0 (Byte)	0 (Signed)	0 (Read)	00010_B
LD.BU (disp23)	dst	0 (Byte)	1 (Unsigned)	0 (Read)	00010_B
LD.H (disp23)	dst	1 (Half-word)	0 (Signed)	0 (Read)	00010_B
LD.HU (disp23)	dst	1 (Half-word)	1 (Unsigned)	0 (Read)	00010_B
LD.W (disp23)	dst	2 (Word)	0 (Signed)	0 (Read)	00010_B
ST.B (disp23)	src	0 (Byte)	0 (Signed)	1 (Write)	00010_B
ST.H (disp23)	src	1 (Half-word)	0 (Signed)	1 (Write)	00010_B
ST.W (disp23)	src	2 (Word)	0 (Signed)	1 (Write)	00010_B
LD.DW (disp23)	dst	3 (Double-word)	0 (Signed)	0 (Read)	00010_B
ST.DW (disp23)	src	3 (Double-word)	0 (Signed)	1 (Write)	00010_B
LDL.W	dst	2 (Word)	0 (Signed)	0 (Read)	00111_B
STC.W	src	2 (Word)	0 (Signed)	1 (Write)	00111_B
CAXI	dst	2 (Word)	1 (Unsigned)	0 (Read)*1	01000_B
SET1	—	0 (Byte)	1 (Unsigned)	0 (Read)*1	01001_B
CLR1	—	0 (Byte)	1 (Unsigned)	0 (Read)*1	01001_B
NOT1	—	0 (Byte)	1 (Unsigned)	0 (Read)*1	01001_B
TST1	—	0 (Byte)	1 (Unsigned)	0 (Read)	01001_B
PREPARE	—	2 (Word)	1 (Unsigned)	1 (Write)	01100_B
DISPOSE	—	2 (Word)	1 (Unsigned)	0 (Read)	01100_B
PUSHSP	—	2 (Word)	1 (Unsigned)	1 (Write)	01101_B
POPSP	—	2 (Word)	1 (Unsigned)	0 (Read)	01101_B
SWITCH	—	1 (Half-word)	0 (Signed)	0 (Read)	10000_B
CALLT	—	1 (Half-word)	1 (Unsigned)	0 (Read)	10001_B
SYSCALL	—	2 (Word)	1 (Unsigned)	0 (Read)	10010_B
CACHE	—	—	—	0/1*2	10100_B
Interrupt (table reference)*3	—	2 (Word)	1 (Unsigned)	0 (Read)	10101_B

Note 1. This exception occurs when the instruction executes a read access.

Note 2. It depends on actual operation.

Note 3. An exception occurs when the table reference interrupt vector is read.

Note: dst: destination register number, src: source register number

(q) RBASE — Reset vector base address register

This register indicates the reset vector address when there is a reset. If the PSW.EBV bit is 0, this vector address is also used as the exception vector address.

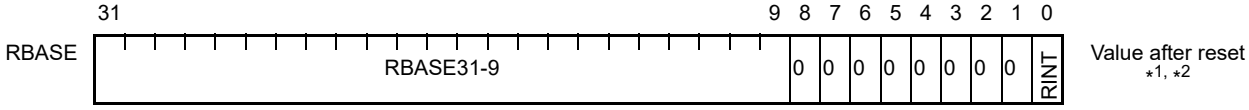


Table 3.23 RBASE Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 9	RBASE31-9	These bits indicate the reset vector when there is a reset. When PSW.EBV = 0, this address is also used as the exception vector. The RBASE8 to 0 bits are implicitly set to 0.	R	*1
8 to 1	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
0	RINT	When the RINT bit is set, the offset address of the exception handler used during interrupt servicing is decremented. See *(1) Direct vector method* in *section 4.5.1 Resets, Exceptions,* and Interrupts in *RH850G3M User's Manual: Software* for the details. This bit is valid when PSW.EBV = 0.	R	*2

Note 1. The reset vector differs depending on the startup area in this product. For details, see **Section 4, Address Space**.

Note 2. The value is 0_B in this product.

(r) EBASE — Exception handler vector address register

This register indicates the exception handler vector address. This register is valid when the PSW.EBV bit is 1.

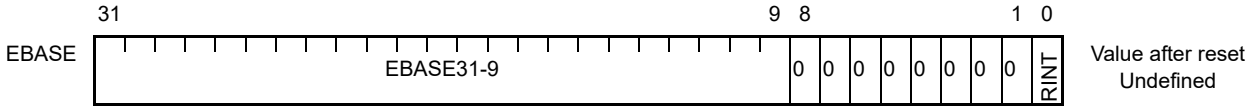


Table 3.24 EBASE Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 9	EBASE31-9	The exception handler routine address is changed to the address resulting from adding the offset address of each exception to the base address specified for this register. The EBASE8 to 0 bits are implicitly set to 0.	R/W	Undefined
8 to 1	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
0	RINT	When the RINT bit is set, the offset address of the exception handler used during interrupt servicing is decremented. See *(1) Direct vector method* in *section 4.5.1 Resets, Exceptions, and Interrupts* in *RH850G3M User's Manual: Software* for the details.	R/W	Undefined

(s) INTBP — Base address of the interrupt handler table register

This register indicates the base address of the address table when the table reference method is selected as the interrupt handler address selection method.

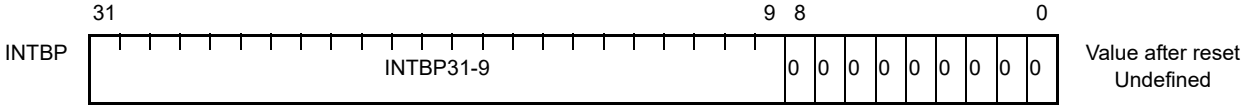


Table 3.25 INTBP Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 9	INTBP31-9	These bits indicate the base pointer address for an interrupt prescribed in the expanded specifications. The value indicated by these bits is the first address in the table used to determine the exception handler when the interrupt prescribed by the expanded specifications (EIINT0 to EIINT511) is acknowledged. The INTBP8 to 0 bits are implicitly set to 0.	R/W	Undefined
8 to 0	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0

(t) PID — Processor ID register

The PID register retains a processor identifier that is unique to the CPU. The PID register is a read-only register.

CAUTION

The PID register indicates information used to identify the incorporated CPU core and CPU core configuration. Usage such that the software behavior varies dynamically according to the PID register information is not assumed.

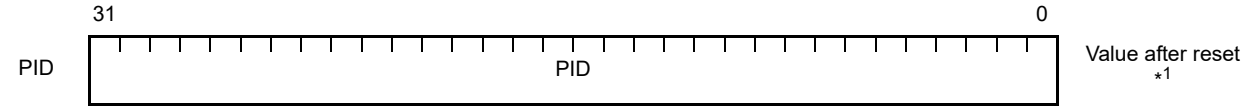


Table 3.26 PID Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 24	PID	Architecture identifier This identifier indicates the architecture of the processor.	R	05_H
23 to 8		Function identifier This identifier indicates the functions of the processor. These bits indicate whether or not functions defined per bit are implemented (1: implemented, 0: not implemented). Bit 23-11 Reserved Bit 10 Double-precision floating-point operation function Bit 9 Single-precision floating-point operation function Bit 8 Memory protection function (MPU)	R	8007_H
7 to 0		Version identifier This identifier indicates the version of the processor.	R	01_H

Note 1. The value is 0580 0701_H in this product.

(u) SCCFG — SYSCALL operation setting register

This register is used to specify operations related to the SYSCALL instruction. Be sure to set an appropriate value to this register before using the SYSCALL instruction.

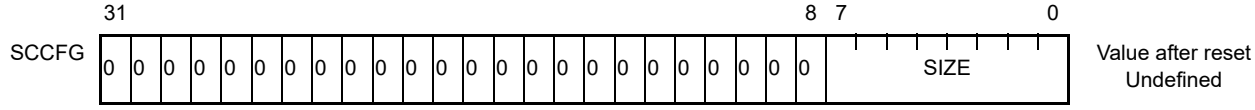


Table 3.27 SCCFG Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 8	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
7 to 0	SIZE	These bits specify the maximum number of entries of a table that the SYSCALL instruction references. The maximum number of entries the SYSCALL instruction references is 1 if SIZE is 0, and 256 if SIZE is 255. By setting the maximum number of entries appropriately in accordance with the number of functions branched by the SYSCALL instruction, the memory area can be effectively used. If a vector exceeding the maximum number of entries is specified for the SYSCALL instruction, the first entry is selected. Place an error processing routine at the first entry.	R/W	Undefined

(v) SCBP — SYSCALL base pointer register

The SCBP register is used to specify a table address of the SYSCALL instruction and generate a target address. Be sure to set an appropriate value to this register before using the SYSCALL instruction.

Be sure to set a word address to the SCBP register.

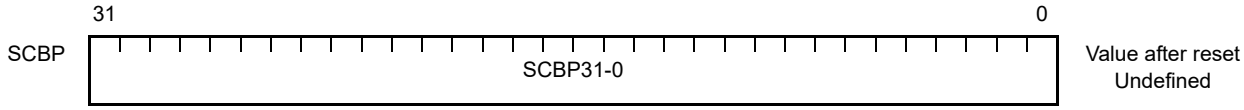


Table 3.28 SCBP Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 2	SCBP31-2	These bits indicate the base pointer address of the SYSCALL instruction. These bits indicate the first address in the table used by the SYSCALL instruction.	R/W	Undefined
1, 0	SCBP1-0	These bits indicate the base pointer address of the SYSCALL instruction. These bits indicate the first address in the table used by the SYSCALL instruction. Always set this bit to 0.	R	0

(w) MCFG0 — Machine configuration register

This register indicates the CPU configuration.

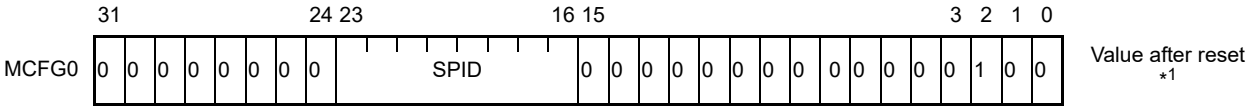


Table 3.29 MCFG0 Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 24	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
23 to 16	SPID	Bit 23 to 18: These are not supported in this product (Reserved for future expansion. Be sure to clear to 0.) Bit 17, 16: These bits indicate the system protection number.	R/W	*2
15 to 3	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
2	—	(Reserved for future expansion. Be sure to set this bit to 1.)	R	1
1, 0	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0

Note 1. The value is 0001 0004_H in CPU1 (PE1) of this product.
Note 2. The value is 01_H in CPU1 (PE1) of this product.

(x) MCTL — Machine control register

This register is used to control the CPU.

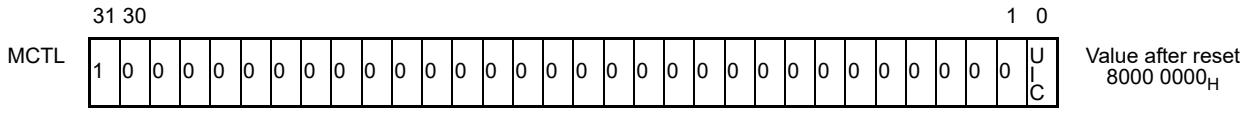


Table 3.30 MCTL Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31	—	(Reserved for future expansion. Be sure to set this bit to 1.)	R	1
30 to 1	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
0	UIC	This bit is used to control the interrupt enable/disable operation in user mode. When this bit is set to 1, executing the EI/DI instruction become possible in user mode.	R/W	0

(3) Interrupt Function Registers

(a) Interrupt function system registers

The interrupt function system registers are read from or written to by using the LDSR or STSR instructions and specifying the system register number, which is made up of a register number and selection ID.

Table 3.31 Interrupt Function System Registers

Register No. (reg.ID, sel.ID)	Symbol	Function	Access Permission
SR7, 1	FPIPR	FPI exception interrupt priority setting	SV
SR10, 2	ISPR	Priority of interrupt being serviced	SV
SR11, 2	PMR	Interrupt priority masking	SV
SR12, 2	ICSR	Interrupt control status	SV
SR13, 2	INTCFG	Interrupt function setting	SV

(1) FPIPR — FPI exception interrupt priority setting register

This register is used to set the interrupt priority of the FPI exception.

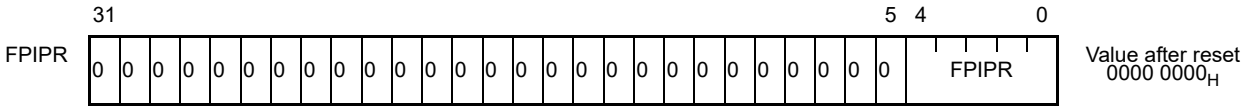


Table 3.32 FPIPR Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 5	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
4 to 0	FPIPR	These bits specify the priority of the floating-point operation exception interrupt (FPI, indicating imprecision). Values from 0 to 15 should be used; the settings 16 and higher are prohibited. FPI exceptions are handled according to this interrupt priority, which is specified in advance. When generated at the same time as another interrupt with the same priority level, the FPI takes priority. **CAUTION** If these bits are set to 16 or a higher value, interrupt masking by the PMR register cannot be applied.	R/W	0

(2) ISPR — Priority of interrupt being serviced register

This register holds the priority of the EIINTn interrupt being serviced. This priority value is then used to perform priority ceiling processing when multiple interrupts are generated.

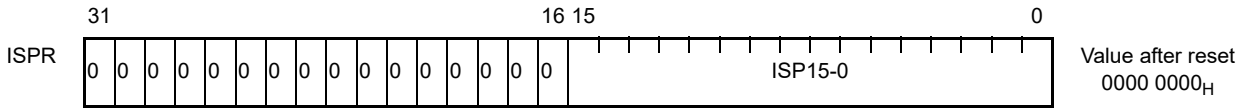


Table 3.33 ISPR Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 16	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
15 to 0	ISP15-0	These bits indicate the acknowledgment status of an EIINTn interrupt with a priority*1 that corresponds to the relevant bit position. 0: An interrupt request for an interrupt whose priority corresponds to the relevant bit position has not been acknowledged. 1: An interrupt request for an interrupt whose priority corresponds to the relevant position is being serviced by the CPU core. The bit positions correspond to the following priority levels: **Bit / Priority** 0 — Priority 0 (highest) 1 — Priority 1 : 14 — Priority 14 15 — Priority 15 When an interrupt request (EIINTn) is acknowledged, the bit corresponding to the acknowledged interrupt request is automatically set to 1. If PSW.EP is 0 when the EIRET instruction is executed, the bit with the highest priority among the ISP15-0 bits that are set (0 is the highest priority) is cleared to 0.*2 While a bit in this register is set to 1, lower priority interrupts (EIINTn) and FPI exception*3 are masked. Priority level judgment is therefore not performed when the system is determining whether to acknowledge an exception, meaning that exceptions will not be acknowledged. See *section 4.1.5 Interrupt Exception Priority and Priority Masking* in *RH850G3M User's Manual: Software* for the details. When performing software-based priority control using the PMR register, be sure to clear this register by using the INTCFG.ISPC bit.	R*4	0

Note 1. See *section 4.1.5 Interrupt Exception Priority and Priority Masking* in *RH850G3M User's Manual: Software* for the details.

Note 2. Interrupt acknowledgment and auto-updating of values when the EIRET instruction is executed are disabled by setting the INTCFG.ISPC bit to 1. We recommend auto-updating, so in normal cases, clear the INTCFG.ISPC bit to 0.

Note 3. Since FPI exceptions have the same level of priority as this interrupt (EIINTn), they are affected by interrupts in the same way as the ISPR. The priority of the FPI exception is set by the FPIPR register.

Note 4. This is R or R/W, depending on the setting of the INTCFG.ISPC bit. It is recommended to use this register as a read-only (R) register.

(3) PMR — Interrupt priority masking register

This register is used to mask the specified interrupt priority.

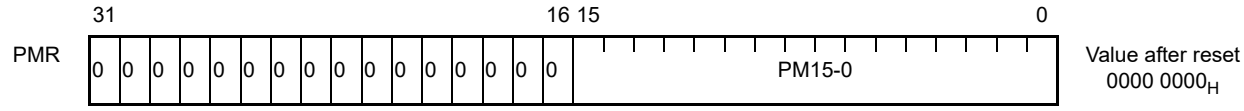


Table 3.34 PMR Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 16	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
15 to 0	PM15-0	These bits mask an interrupt request with a priority level that corresponds to the relevant bit position. 0: Servicing of an interrupt request with a priority that corresponds to the relevant bit position is enabled. 1: Servicing of an interrupt request with a priority that corresponds to the relevant bit position is disabled. The bit positions correspond to the following priority levels: **Bit** / **Priority** 0 / Priority 0 (highest) 1 / Priority 1 : / : 14 / Priority 14 15 / Priority 15 (lowest) While a bit in this register is set to 1, interrupts (EIINTn) with the priority corresponding to that bit and FPI exception*1 are masked. Priority level judgment is therefore not performed when the system is determining whether to acknowledge an exception, meaning that exceptions will not be acknowledged*2.	R/W	0

Note 1. Since a FPI exception is specified as the same level of priority as that of an interrupt (EIINTn), it is affected by the PMR like interrupts. The priority of FPI exception is set by the FPIPR register.

Note 2. Specify the masks by setting the bits to 1 in order from the lowest-priority bit. For example, $FF00_H$ can be set, but $F0F0_H$ or $00FF_H$ cannot.

(4) ICSR — Interrupt control status register

This register indicates the interrupt control status in the CPU.

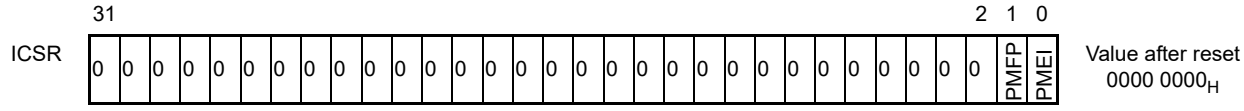


Table 3.35 ICSR Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 2	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
1	PMFP	This bit indicates that an FPI exception with the priority level masked by the PMR register exists.	R	0
0	PMEI	This bit indicates that an interrupt (EIINTn) with the priority level masked by the PMR register exists	R	0

(5) INTCFG — Interrupt function setting register

This register is used to specify settings related to the CPU's internal interrupt function.

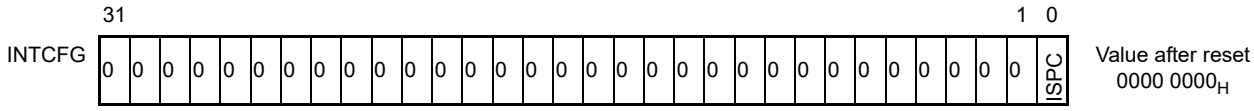


Table 3.36 INTCFG Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 1	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
0	ISPC	This bit changes how the ISPR register is written. 0: The ISPR register is automatically updated. Updates triggered by the program (via execution of LDSR instruction) are ignored. 1: The ISPR register is not automatically updated. Updates triggered by the program (via execution of LDSR instruction) are performed. If this bit is cleared to 0, the bits of the ISPR register are automatically set to 1 when an interrupt (EIINTn) is acknowledged, and cleared to 0 when the EIRET instruction is executed. In this case, the bits are not updated by an LDSR instruction executed by the program. If this bit is set to 1, the bits of the ISPR register are not updated by the acknowledgement of an interrupt (EIINTn) or by execution of the EIRET instruction. In this case, the bits can be updated by an LDSR instruction executed by the program. In normal cases, the ISPC bit should be cleared. When performing software-based priority control, however, set this bit to 1 and perform priority control by using the PMR register.	R/W	0

(4) FPU Function Registers

(a) Floating-point operation registers

The FPU uses the CPU general-purpose registers (r0 to r31). There are no register files used only for floating-point operations.

- Single-precision floating-point instruction
  Thirty-two 32-bit registers can be specified. These general-purpose registers correspond to r0 to r31.
- Double-precision floating-point instruction
  Sixteen 64-bit registers can be specified. Paired general-purpose registers are used as register pairs ({r1, r0}, {r3, r2} … {r31, r30}). Each register pair is specified in the instruction format with an even numbered register. Because r0 is a zero register (always holds “0”), in principle {r1, r0} should not be used by a double-precision floating-point instruction.

(b) Floating-point system registers

The FPU can use the following system registers to control floating-point operation. Reading from/ writing to a floating-point system register is performed by specifying the system register number, which consists of the register number and selection identifier, with the LDSR or STSR instructions. For details of the registers, see *section 3.4.2 Floating-Point Function System Registers* in *RH850G3M User’s Manual: Software*.

Table 3.37 FPU Function System Registers

Register No. (reg.ID, sel.ID)	Symbol	Function	Access Permission
SR6, 0	FPSR	Floating-point operation setting/status	CU0 and SV
SR7, 0	FPEPC	Floating-point operation exception program counter	CU0 and SV
SR8, 0	FPST	Floating-point status	CU0
SR9, 0	FPCC	Floating-point operation comparison result	CU0
SR10, 0	FPCFG	Floating-point function setting	CU0
SR11, 0	FPEC	Floating-point operation exception control	CU0 and SV

(5) MPU function registers

(a) MPU function system registers

The MPU function system registers are read from or written to by using the LDSR or STSR instructions and specifying the system register number, which is made up of a register number and selection ID.

Table 3.38 MPU Function System Registers (1/2)

Register No. (reg.ID, sel.ID)	Symbol	Function	Access Permission
SR0, 5	MPM	Memory protection operation mode setting	SV
SR1, 5	MPRC	MPU region control	SV
SR4, 5	MPBRGN	MPU base region number	SV
SR5, 5	MPTRGN	MPU end region number	SV
SR8, 5	MCA	Memory protection setting check address	SV
SR9, 5	MCS	Memory protection setting check size	SV
SR10, 5	MCC	Memory protection setting check command	SV
SR11, 5	MCR	Memory protection setting check result	SV
SR0, 6	MPLA0	Protection area minimum address	SV
SR1, 6	MPUA0	Protection area maximum address	SV
SR2, 6	MPAT0	Protection area attribute	SV
SR4, 6	MPLA1	Protection area minimum address	SV
SR5, 6	MPUA1	Protection area maximum address	SV
SR6, 6	MPAT1	Protection area attribute	SV
SR8, 6	MPLA2	Lower address of the protection area	SV
SR9, 6	MPUA2	Protection area maximum address	SV
SR10, 6	MPAT2	Protection area attribute	SV
SR12, 6	MPLA3	Protection area minimum address	SV
SR13, 6	MPUA3	Protection area maximum address	SV
SR14, 6	MPAT3	Protection area attribute	SV
SR16, 6	MPLA4	Protection area minimum address	SV
SR17, 6	MPUA4	Protection area maximum address	SV
SR18, 6	MPAT4	Protection area attribute	SV
SR20, 6	MPLA5	Protection area minimum address	SV
SR21, 6	MPUA5	Protection area maximum address	SV
SR22, 6	MPAT5	Protection area attribute	SV
SR24, 6	MPLA6	Protection area minimum address	SV
SR25, 6	MPUA6	Protection area maximum address	SV
SR26, 6	MPAT6	Protection area attribute	SV
SR28, 6	MPLA7	Protection area minimum address	SV
SR29, 6	MPUA7	Protection area maximum address	SV
SR30, 6	MPAT7	Protection area attribute	SV
SR0, 7	MPLA8	Protection area minimum address	SV
SR1, 7	MPUA8	Protection area maximum address	SV
SR2, 7	MPAT8	Protection area attribute	SV
SR4, 7	MPLA9	Protection area minimum address	SV
SR5, 7	MPUA9	Protection area maximum address	SV

Table 3.38 MPU Function System Registers (2/2)

Register No. (reg.ID, sel.ID)	Symbol	Function	Access Permission
SR6, 7	MPAT9	Protection area attribute	SV
SR8, 7	MPLA10	Protection area minimum address	SV
SR9, 7	MPUA10	Protection area maximum address	SV
SR10, 7	MPAT10	Protection area attribute	SV
SR12, 7	MPLA11	Protection area minimum address	SV
SR13, 7	MPUA11	Protection area maximum address	SV
SR14, 7	MPAT11	Protection area attribute	SV
SR16, 7	MPLA12	(Not implemented. A value of 0 is returned when read and writing is ignored.)	SV
SR17, 7	MPUA12	(Not implemented. A value of 0 is returned when read and writing is ignored.)	SV
SR18, 7	MPAT12	(Not implemented. A value of 0 is returned when read and writing is ignored.)	SV
SR20, 7	MPLA13	(Not implemented. A value of 0 is returned when read and writing is ignored.)	SV
SR21, 7	MPUA13	(Not implemented. A value of 0 is returned when read and writing is ignored.)	SV
SR22, 7	MPAT13	(Not implemented. A value of 0 is returned when read and writing is ignored.)	SV
SR24, 7	MPLA14	(Not implemented. A value of 0 is returned when read and writing is ignored.)	SV
SR25, 7	MPUA14	(Not implemented. A value of 0 is returned when read and writing is ignored.)	SV
SR26, 7	MPAT14	(Not implemented. A value of 0 is returned when read and writing is ignored.)	SV
SR28, 7	MPLA15	(Not implemented. A value of 0 is returned when read and writing is ignored.)	SV
SR29, 7	MPUA15	(Not implemented. A value of 0 is returned when read and writing is ignored.)	SV
SR30, 7	MPAT15	(Not implemented. A value of 0 is returned when read and writing is ignored.)	SV

Note 1. The number of the MPLAn, MPUAn, and MPATn registers incorporated is dependent on the product. The number of the protection area in this product is 12 (n = 0 to 11).

(1) MPM — Memory protection operation mode register

The memory protection mode register is used to define the basic operating state of the memory protection function.

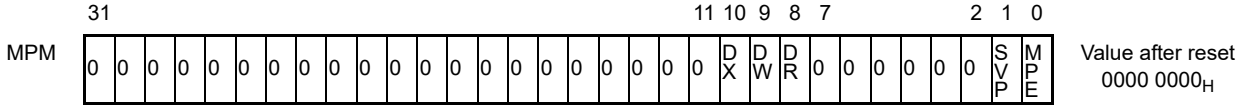


Table 3.39 MPM Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 11	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
10	DX	This bit is not supported in this product. (DX is reserved for future expansion. Be sure to clear to 0.)	R	0
9	DW	This bit is not supported in this product. (DW is reserved for future expansion. Be sure to clear to 0.)	R	0
8	DR	This bit is not supported in this product. (DR is reserved for future expansion. Be sure to clear to 0.)	R	0
7 to 2	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
1	SVP	In SV mode. (when PSW.UM = 0), this bit is used to specify whether to restrict access according to the SX, SW, and SR bits of the MPAT register for each protection area.*1 0: As usual, implicitly enable all access in SV mode. 1: Restrict access according to the SX, SW, and SR bits even in SV mode.*2	R/W	0
0	MPE	This bit is used to specify whether to enable or disable MPU function. 0: Disable 1: Enable	R/W	0

Note 1. If the SVP bit is set to 1, access will be restricted in accord with the setting for each protection area, even in SV mode. Therefore, specify the protection area before setting the SVP bit to prevent restriction of access by the program itself.

Note 2. If access is restricted in SV mode, execution of MDP exceptions or the MIP exception handling itself might not be possible depending on the settings. Be careful to specify settings so that access to the memory area necessary for the exception handler and exception handling is permitted.

(2) MPRC — MPU region control register

Bits used to perform special memory protection function operations are located in this register.

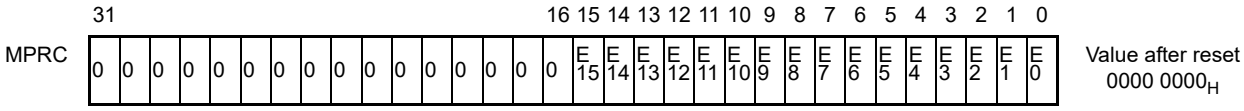


Table 3.40 MPRC Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 16	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
15 to 12	E15-E12	These are not supported in this product. (Reserved for future expansion. Be sure to clear to 0.)	R	0
11 to 0	E11-E0	These are the enable bits for each protection area. Bit En is a copy of bit MPATn.E (where n = 11 to 0). The number of the protection areas in this product is 12.	R/W	0

(3) MPBRGN — MPU base region register

This register indicates the minimum usable MPU area number.

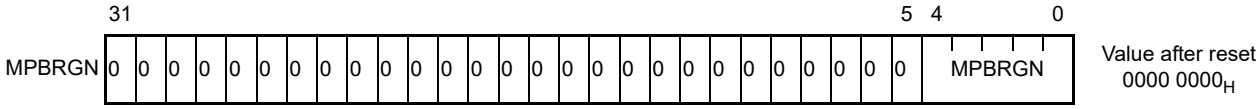


Table 3.41 MPBRGN Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 5	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
4 to 0	MPBRGN	These bits indicate the smallest number of an MPU area. These bits are always read as 0.	R	0

(4) MPTRGN — MPU end region register

This register indicates the maximum usable MPU area number + 1.

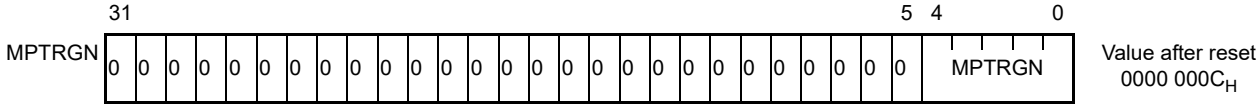


Table 3.42 MPTRGN Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 5	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
4 to 0	MPTRGN	These bits indicate the largest number of an MPU area plus one. These bits always indicate the maximum number of MPU areas that the hardware can support. The number of the protection areas in this product is 12.	R	12

(5) MCA — Memory protection setting check address register

This register is used to specify the base address of the area for which a memory protection setting check is to be performed.

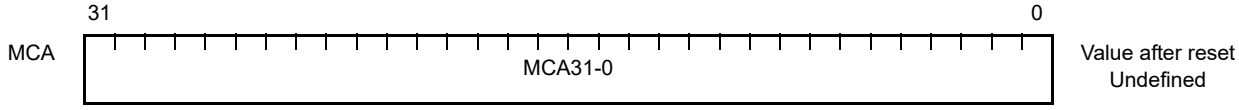


Table 3.43 MCA Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 0	MCA31-0	These bits are used to specify the start address of the memory area subject to a memory protection setting check in bytes.	R/W	Undefined

(6) MCS — Memory protection setting check size register

This register is used to specify the size of the area for which a memory protection setting check is to be performed.

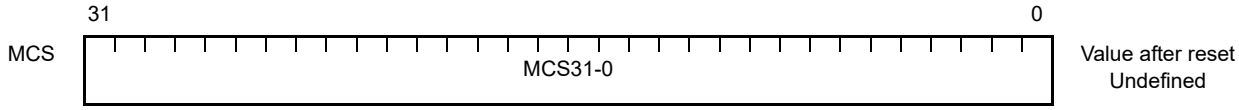


Table 3.44 MCS Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 0	MCS31-0	These bits specify the size of the target area to specify the size of memory area subject to a memory protection setting check in bytes. These bits specify the size in bytes of a memory area that is subject to a memory protection setting. Checking in areas below the address value in the MCS register is not possible because the specified size is handled as an unsigned integer. Do not set 0000 0000_H in the MCS register.	R/W	Undefined

(7) MCC — Memory protection setting check command register

This command register is used to start a memory protection setting check.

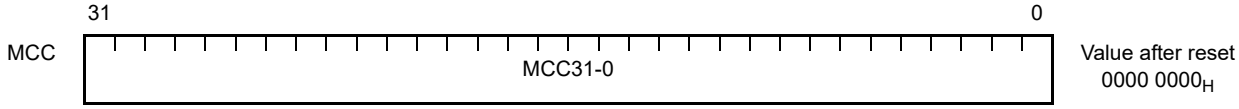


Table 3.45 MCC Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 0	MCC31-MCC0	When any value is written to the MCC register, a memory protection setting check starts. Setting the MCA and MCS registers and then writing to this register leads to storage of the result of checking in the MCR register. Since writing any value to this register starts the check, doing so does not require any extra registers when r0 is used as a source register. The result of checking is reflected in MCR according to any area setting regardless of the setting of the PSW.UM bit. The value read from the MCC register is always 0000 0000_H.	R/W	0

(8) MCR — Memory protection setting check result register

This register is used to store the results of a memory protection setting check.

Be sure to clear bit 31 to 9, 7, and 6.

CAUTIONS

1. **If the area for which checking is specified crosses 0000 0000$_H$, the area is judged to have been specified incorrectly, and the MCR.OV bit is set to 1. For this reason, in access to the results of checking, check the MCR.OV bit and confirm that the result is valid (OV = 0) before using any results of checking.**
2. **With the default settings (for MPM.DX, MPM.DW, MPM.DR) of 1, the correct result may not be obtained. If the specified default is to be enabled, do not use the memory protection setting check function.**

	31																					9	8	7	6	5	4	3	2	1	0	
MCR	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	OV	0	0	SXE	SWE	SRE	UXE	UWE	URE	Value after reset Undefined

Table 3.46 MCR Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 9	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
8	OV	If the specified area includes 0000 0000$_H$ or 7FFF FFFF$_H$, 1 is stored in this bit. In other cases, 0 is stored in this bit.	R/W	Undefined
7, 6	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
5	SXE	If the specified area is contained within one protection area and execution is permitted for that area in supervisor mode, 1 is stored in this bit. In other cases, 0 is stored in this bit.	R/W	Undefined
4	SWE	If the specified area is contained within one protection area and writing to that area is permitted in supervisor mode, 1 is stored in this bit. In other cases, 0 is stored in this bit.	R/W	Undefined
3	SRE	If the specified area is contained within one protection area and reading from that area is permitted in supervisor mode, 1 is stored in this bit. In other cases, 0 is stored in this bit.	R/W	Undefined
2	UXE	If the specified area is contained within one protection area and execution is permitted for that area in user mode, 1 is stored in this bit. In other cases, 0 is stored in this bit.	R/W	Undefined
1	UWE	If the specified area is contained within one protection area and writing to that area is permitted in user mode, 1 is stored in this bit. In other cases, 0 is stored in this bit.	R/W	Undefined
0	URE	If the specified area is contained within one protection area and reading from that area is permitted in user mode, 1 is stored in this bit. In other cases, 0 is stored in this bit.	R/W	Undefined

(9) MPLAn — Protection area minimum address register

These registers indicate the minimum address of area n (where n = 0 to 11). The number of the protection areas in this product is 12.

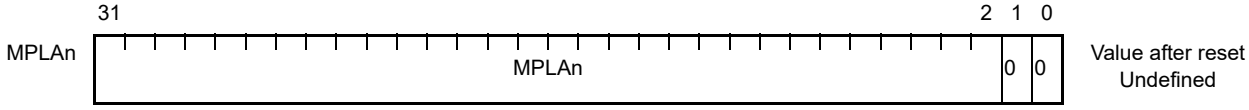


Table 3.47 MPLAn Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 2	MPLA31-2	These bits indicate the minimum address of area n. The MPLA1-0 bits are implicitly set to 0.	R/W	Undefined
1, 0	—	Reserved for future expansion. Be sure to clear these bits to 0.	R	0

(10) MPUAn — Protection area maximum address register

These registers indicate the maximum address of area n (where n = 0 to 11). The number of the protection areas in this product is 12.

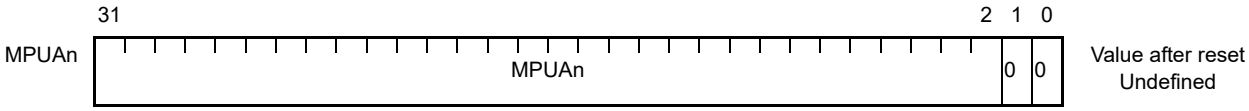


Table 3.48 MPUAn Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 2	MPUA31-2	These bits indicate the maximum address of area n. The MPUA1-0 bits are implicitly set to 1.	R/W	Undefined
1, 0	—	Reserved for future expansion. Be sure to clear these bits to 0.	R	0

(11) MPATn — Protection area attribute register

These registers indicate the attributes of area n (where n = 0 to 11). The number of the protection areas in this product is 12.

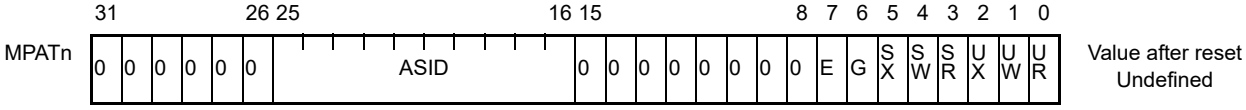


Table 3.49 MPATn Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 26	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
25 to 16	ASID	These bits indicate the ASID value to be used as the area match condition.	R/W	Undefined
15 to 8	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
7	E	This bit indicates whether area n is enabled or disabled. 0 : Area n is disabled. 1 : Area n is enabled.	R/W	0
6	G	0: ASID match is the condition. 1: ASID match is not the condition. When this bit is 0, the condition of the area match is MPATn.ASID = ASID.ASID. When this bit is 1, the area match of the values of MPATn.ASID and ASID.ASID is not the condition.	R/W	Undefined
5	SX	This bit indicates the execution privilege in supervisor mode.*1 0 : Execution is disabled. 1 : Execution is enabled.	R/W	Undefined
4	SW	This bit indicates the write permission in supervisor mode.*1 0 : Writing is disabled. 1 : Writing is enabled.	R/W	Undefined
3	SR	This bit indicates the read permission in supervisor mode*1 0 : Reading is disabled. 1 : Reading is enabled.	R/W	Undefined
2	UX	This bit indicates the execution privilege in user mode. 0 : Execution is disabled. 1 : Execution is enabled.	R/W	Undefined
1	UW	This bit indicates the write permission in user mode. 0 : Writing is disabled. 1 : Writing is enabled.	R/W	Undefined
0	UR	This bit indicates the read permission in user mode. 0 : Reading is disabled. 1 : Reading is enabled.	R/W	Undefined

Note 1. If access is restricted in SV mode, execution of MDP exceptions or the MIP exception handling itself might not be possible depending on the settings. Be careful to specify settings so that access to the memory area necessary for the exception handler and exception handling is permitted.

(6) Cache Operation Function registers

(a) Cache control function system registers

The cache control function system registers are read from or written to by using the LDSR or STSR instructions and specifying the system register number, which is made up of a register number and selection ID.

Table 3.50 Cache Control System Registers

Register No. (reg.ID, sel.ID)	Symbol	Function	Access Permission
SR16, 4	ICTAGL	Instruction cache tag Lo access	SV
SR17, 4	ICTAGH	Instruction cache tag Hi access	SV
SR18, 4	ICDATL	Instruction cache data Lo access	SV
SR19, 4	ICDATH	Instruction cache data Hi access	SV
SR24, 4	ICCTRL	Instruction cache control	SV
SR26, 4	ICCFG	Instruction cache configuration	SV
SR28, 4	ICERR	Instruction cache error	SV

(1) ICTAGL — Instruction cache tag Lo access register

This register is used for CIST and CILD instructions for the instruction cache. This register holds values to be stored in the tag RAM of the instruction cache by the execution of CIST instructions and values read from the tag RAM of the instruction cache by the execution of CILD instructions.

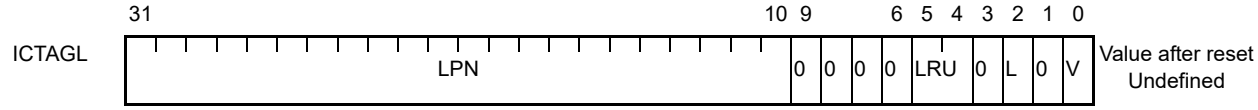


Table 3.51 ICTAGL Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 10	LPN	These bits hold the values of bits 24 to 11, i.e. the physical page numbers. When writing, always write 0 to bits 31 to 25 and to bit 10.	R/W	Undefined
9 to 6	—	Reserved for future expansion. When writing, always write 0 to these bits.	R	0
5, 4	LRU	These bits indicate the LRU information of the specified cache line. The CIST instruction cannot be used to change the LRU information to desired values.	R/W	Undefined
3	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
2	L	This bit holds the lock information.	R/W	Undefined
1	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
0	V	This bit retains whether the specified cache line is enabled or disabled.	R/W	Undefined

(2) ICTAGH — Instruction cache tag Hi access register

This register is used for a CIST or CILD instruction for the instruction cache. This register stores the value to be stored in the instruction cache tag RAM on CIST execution and the value read from the instruction cache tag RAM on CILD execution.

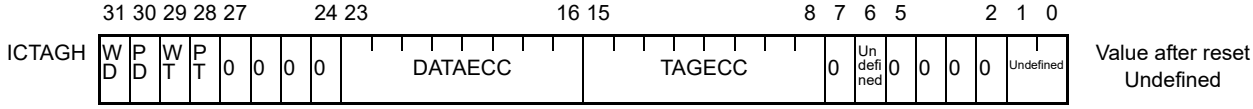


Table 3.52 ICTAGH Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31	WD	When this bit is set to 1, executing a CIST instruction updates the cache data RAM.	R/W	Undefined
30	PD	When this bit is set to 1, executing a CIST instruction writes the value in the DATAECC field of this register to the ECC field of the data RAM.	R/W	Undefined
29	WT	This bit updates the cache tag RAM when this bit is set to 1 before CIST execution.	R/W	Undefined
28	PT	This bit writes the value in TAGECC field to ECC for tag RAM when this bit is set to 1 before CIST execution.	R/W	Undefined
27 to 24	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
23 to 16	DATAECC	These bits retain ECC of the data RAM.	R/W	Undefined
15 to 8	TAGECC	These bits retain ECC of the tag RAM. Be sure to clear bit 15 to 0.	R/W	Undefined
7	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
6	—	(Reserved for future expansion. Be sure to clear to 0.)	R	Undefined
5 to 2	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
1, 0	—	(Reserved for future expansion. Be sure to clear to 0.)	R	Undefined

(3) ICDATL — Instruction cache data Lo access register

This register is used for a CIST or CILD instruction for the instruction cache. This register stores the value to be stored in the instruction cache data RAM on CIST execution and the value read from the instruction cache data RAM on CILD execution.

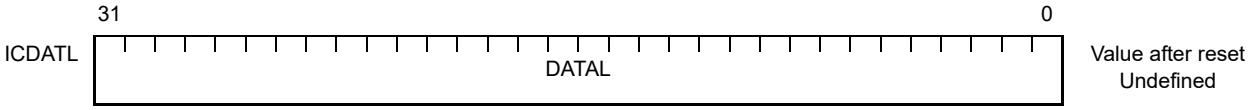


Table 3.53 ICDATL Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 0	DATAL	These bits retain the values of bits 31 to 0 or of bit 95 to 64 from the instruction data of the block within the specified cache line. The offset of the index specifies the target range of bit numbers. Index offset = 0000: bits 31 to 0 Index offset = 1000: bits 95 to 64	R/W	Undefined

(4) ICDATH — Instruction cache data Hi access register

This register is used for a CIST or CILD instruction for the instruction cache. This register stores the value to be stored in the instruction cache data RAM on CIST execution and the value read from the instruction cache data RAM on CILD execution.

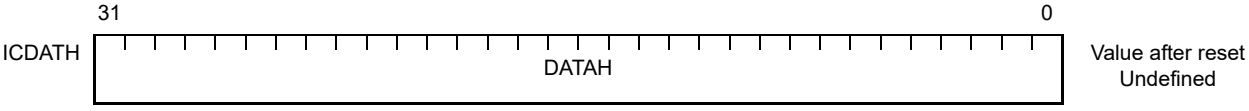


Table 3.54 ICDATH Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 0	DATAH	These bits retain the values of bit 63 to 32 or those of bit 127 to 96 from the instruction data of the block within the specified cache line. The index offset specifies the bit number to be retained. Index offset = 0000: bit 63 to 32 Index offset = 1000: bit 127 to 96	R/W	Undefined

(5) ICCTRL — Instruction cache control register

This register controls the instruction cache.

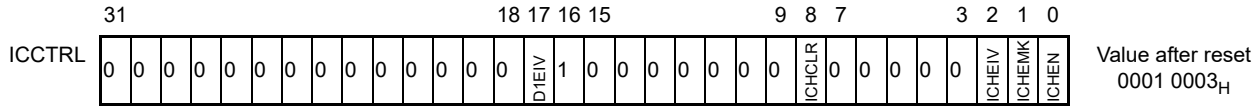


Table 3.55 ICCTRL Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 18	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
17	D1EIV	This bit selects the operation in response to 1-bit errors in the data RAM. 0: The error is corrected and then processing continues, but the address of the entry that had an error is retained. 1: The error is not corrected, the entry is cleared, and fetching is repeated. This bit is read as the previous value until the setting is actually reflected in the instruction cache.	R/W	0
16	—	(Reserved for future expansion. Be sure to set to 1.)	R	1
15 to 9	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
8	ICHCLR	Setting this bit to 1 selects clearing of the whole instruction cache in a single operation. After this bit has been set to 1, it will be read as 1 until clearing is completed. The bit is cleared to 0 once clearing of the cache is completed.	R/W	0
7 to 3	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
2	ICHEIV	Setting this bit to 1 allows the instruction cache to be automatically disabled (ICHEN bit to be cleared to 0) when a cache error occurs.	R/W	0
1	ICHEMK	Setting this bit to 1 selects masking of cache error exception notifications for the CPU when cache errors are encountered.	R/W	1
0	ICHEN	This bit disables or enables the instruction cache. 0: Instruction cache is disabled. 1: Instruction cache is enabled. This bit is read as the previous value until the setting is actually reflected in the instruction cache.	R/W	1

(6) ICCFG — Instruction cache configuration register

This register shows the configuration of the instruction cache.

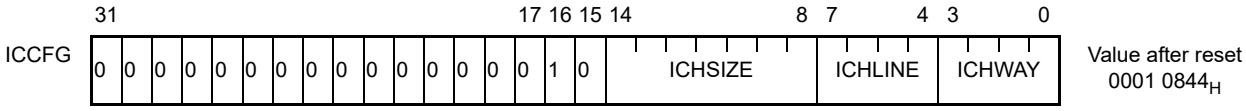


Table 3.56 ICCFG Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 17	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
16	—	(Reserved for future expansion. Be sure to set to 1.)	R	1
15	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
14 to 8	ICHSIZE	These bits indicate the capacity (in Kbytes) of the instruction cache. The setting is 8 Kbytes in this product. 000 1000: 8 Kbytes	R	08_H
7 to 4	ICHLINE	These bits indicate the number of lines per 1 way in the instruction cache. The setting is 128 lines in this product. 0100: 128 lines	R	4_H
3 to 0	ICHWAY	These bits indicate the number of ways of the instruction cache. The setting is 4 ways in this product. 0100: 4 ways	R	4_H

(7) ICERR — Instruction cache error register

This register stores cache error data of the instruction cache.

Once the ICHERR bit is set to 1, subsequent cache error data is not stored in this register until the ICHERR bit is cleared to 0.

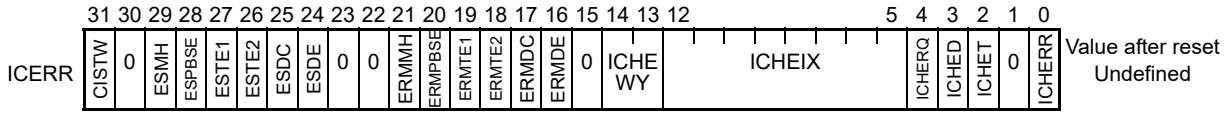


Table 3.57 ICERR Register Contents (1/2)

Bit Position	Bit Name	Function	R/W	Value after Reset
31	CISTW	This bit is set to indicate that the destination way specified for a CISTI instruction was in error. Although the entry information is overwritten so that writing is completed, the V bit will be cleared the next time the cache line is read (i.e. reading will be judged to have missed the cache). However, setting of this bit is not accompanied by an exception for the CPU.	R/W	0
30	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
29	ESMH	Error status: multi-hit	R/W	Undefined
28	ESPBSE	Error status: way error	R/W	Undefined
27	ESTE1	Error status: 1-bit error in the tag RAM	R/W	Undefined
26	ESTE2	Error status: 2-bit error in the tag RAM	R/W	Undefined
25	ESDC	Error status: 1-bit correction in the data RAM	R/W	Undefined
24	ESDE	Error status: 2-bit error in the data RAM	R/W	Undefined
23, 22	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
21	ERMMH	Error exception notification mask: multi-hit	R/W	0
20	ERMPBSE	Error exception notification mask: way error	R/W	0

Table 3.57 ICERR Register Contents (2/2)

Bit Position	Bit Name	Function	R/W	Value after Reset
19	ERMTE1	Error exception notification mask: 1-bit error in the tag RAM	R/W	0
18	ERMTE2	Error exception notification mask: 2-bit error in the tag RAM	R/W	0
17	ERMDC	Error exception notification mask: 1-bit correction in the data RAM	R/W	0
16	ERMDE	Error exception notification mask: 2-bit error in the data RAM	R/W	0
15	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
14, 13	ICHEWY	These bits retain a way number where a cache error occurs.	R/W	Undefined
12 to 5	ICHEIX	These bits retain a cache index where a cache error occurs.	R/W	Undefined
4	ICHERQ	Setting of this bit to 1 indicates that the CPU is being notified of a cache error exception. If cache error exceptions are masked, however, the CPU is not notified of an exception even when this bit is set to 1.	R/W	0
3	ICHED	This bit indicates that an error occurs in the data RAM.	R/W	0
2	ICHET	This bit indicates that an error occurs in the tag RAM.	R/W	0
1	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
0	ICHERR	This bit is set to 1 when a cache error occurs.	R/W	0

(7) Data Buffer Operation Function Registers

(a) Data buffer control function system registers

The data buffer control function system registers are read from or written to by using the LDSR or STSR instructions and specifying the system register number, which is made up of a register number and selection ID.

Table 3.58 Data Buffer Operation Function Registers

Register No. (reg.ID, sel.ID)	Symbol	Function	Access Permission
SR24,13	CDBCR	Data buffer control register	SV

(1) CDBCR — Data Buffer Control Register

This register controls the data buffer.

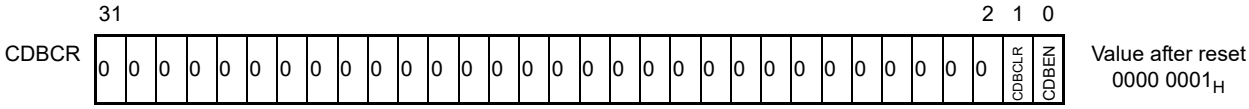


Table 3.59 CDBCR Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 2	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
1	CDBCLR	When this bit is set to 1, the data buffer is cleared at a time. This bit is always read as 0.	W	0
0	CDBEN	This bit disables or enables the data buffer. 0: Data buffer is disabled. 1: Data buffer is enabled.	R/W	1

3.2.2 Instruction Cache and Data Buffer

3.2.2.1 Features

An 8-Kbyte and 4-way set-associative instruction cache is mounted between the CPU1 and the code flash. The instruction cache and the code flash are connected to each other via a 128-bit dedicated bus to minimize penalties caused by a cache miss-hit. Also a data buffer is mounted between the CPU1 and the code flash to achieve high-speed data access. The 32-MB area from 0000 0000_H to 01FF $FFFF_H$ in the address space is intended for the instruction cache and data buffer.

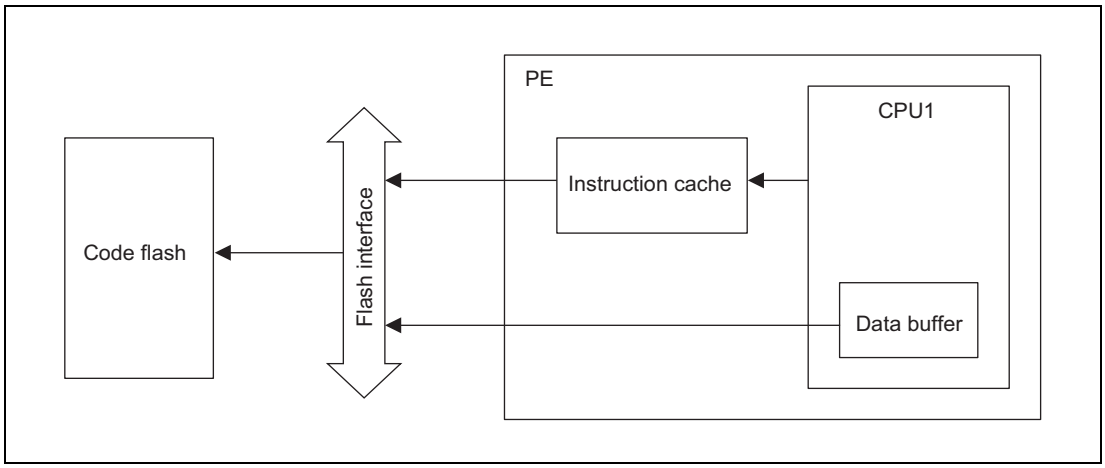


Figure 3.2 Instruction Cache and Data Buffer

3.2.2.2 Instruction Cache Function

The 8-Kbyte and 4-way set-associative cache includes four ways consisting of 128-entry blocks of four words per line, amounting to 8-Kbyte capacity in total. The ways are divided into two groups, way group 0 consisting of way 0 and way 1 and way group 1 consisting of way 2 and way 3. The way group can be selected and used by decoding of the address information of the access destination. If a cache error occurs, each line is refilled by a replace algorithm using LRU. CPU instruction fetches from the code flash are performed via the instruction cache.

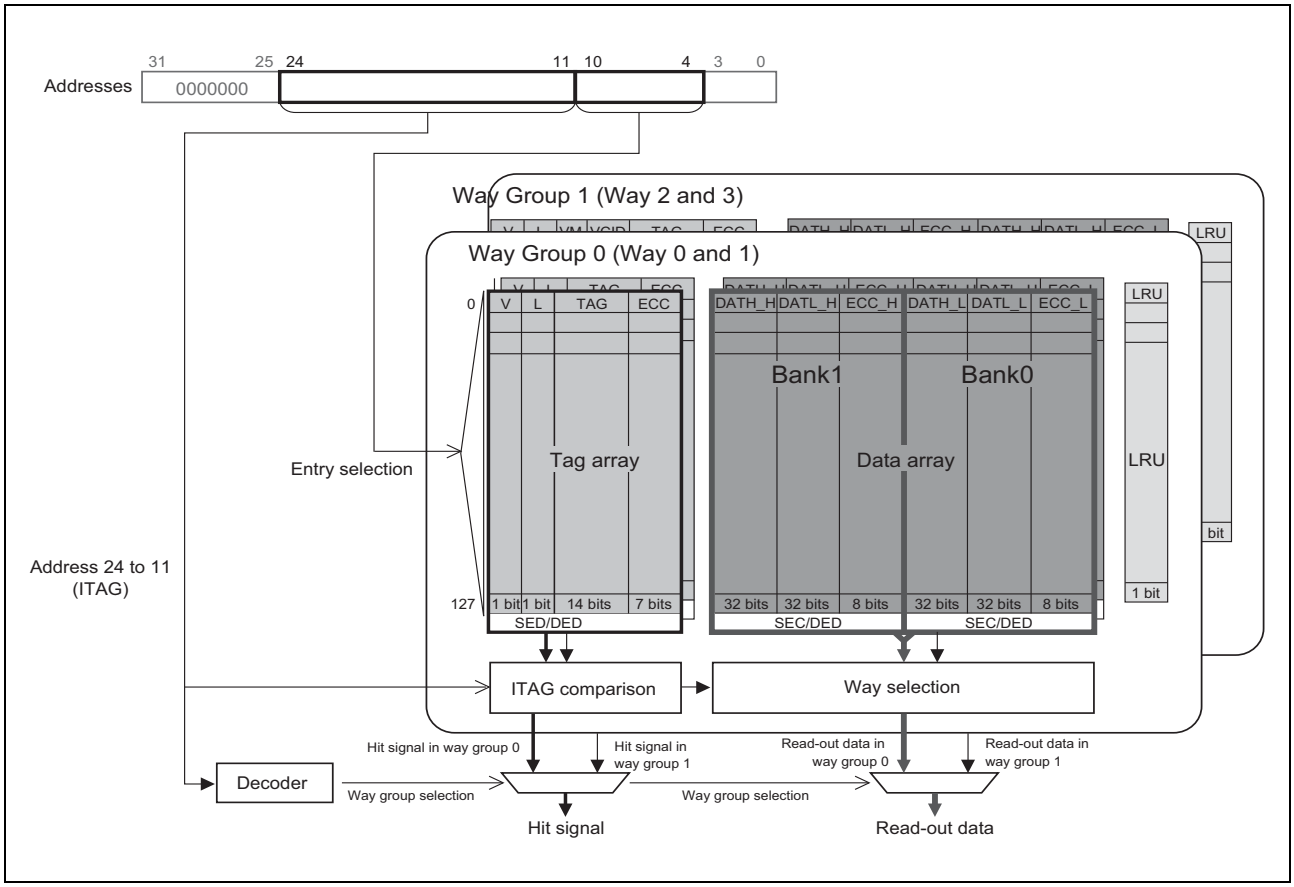


Figure 3.3 Instruction Cache Configuration

Tag Array

V bit	This bit indicates whether valid data is stored in the cache line. Setting of this bit to 1 makes the cache line data valid. The V bit is initialized to 0 by reset.
L bit	This bit indicates whether a cache line is locked or not. Setting of this bit to 1 locks the cache line and it cannot be replaced with new data. The L bit is valid only when the V bit is 1, and it is not initialized by reset.
TAG	Among 32 bits in the operable addresses of the data line to be cached, bits 24 to 11 are stored in this bit. The TAG bit is not initialized by reset.
ECC	The ECC of the tag array is stored in this bit. The ECC bit is not initialized by reset.

Data Array

DATH_H, DATL_H, DATH_L, DATL_L	The 128-bit cache line data is stored per 32 bits as follows: the bits [127:96], [95:64], [63:32], [31:0] in the DATH_H, DATL_H, DATH_L, and DATL_L, respectively. In the CIST or CILD operation in response to a cache instruction, the ICDATH register is used for DATH_H and DATH_L and ICDATL is used for DATL_H and DATL_L.
ECC_H, ECC_L	The ECC of the data in bits [127:64] and [63:0] are stored in ECC_H and ECC_L, respectively.

LRU

LRU	The LRU information in the same way group is stored in this data array. The LRU is initialized by reset.

CAUTION

When an instruction is fetched from an applicable line after issuing a CIST instruction for writing test data to the tag array of the instruction cache, the tag information must be written in a group unit. For example, when writing tag information to a line on the way 0 side of way group 0, also write tag information for the same line to way 1, and then execute the instruction fetch

- When writing to way group 0 (way 0 and way 1), write a value such that the exclusive OR of the ICTAGL.LPN bit is 0.
- When writing to way group 1 (way 2 and way 3), write a value such that the exclusive OR of the ICTAGL.LPN bit is 1.

Fetching an instruction after a value that does not follow the above rule has been written to the tag array causes a way error and setting of the ICERR.ESPBSE bit to 1. Fetching an instruction after the same tag information has been written to the same lines of both ways in a way group causes a multi-hit error and setting of the ICERR.ESMH bit to 1.

3.2.2.3 Data Buffer Function

The four-line buffer with 128 bits per line is mounted as a data buffer. The data of 128 bits per line read from the code flash is stored in the data buffer. The data is read out from the data buffer after the next access to the same address, so the code flash is not accessed again.

3.2.3 Inter-Processor Interrupts

Registers (IPIR_CHn) for interrupt communication between PEs are provided for four channels.

IPIR_CH0 to IPIR_CH3 are assigned to CH0 to CH3 of user interrupt (EIINT). An interrupt for specific PEs (including own PE) can be requested by manipulating bits corresponding to respective PEs.

3.2.3.1 Inter-Processor Interrupt Control Registers

This register is located in the CPU peripheral of each PE. IPIR_CH0 to IPIR_CH3 registers are assigned to each PE which are accessible only from their own PE.

Table 3.60 List of Registers

Register Symbol	Register Name	R/W	Value after Reset	Access Size				Address
				1	8	16	32	
IPIR_CH0	Inter-PE interrupt register 0	R/W	0000 0000$_H$	√	√	√	√	FFFE EC80$_H$
IPIR_CH1	Inter-PE interrupt register 1	R/W	0000 0000$_H$	√	√	√	√	FFFE EC84$_H$
IPIR_CH2	Inter-PE interrupt register 2	R/W	0000 0000$_H$	√	√	√	√	FFFE EC88$_H$
IPIR_CH3	Inter-PE interrupt register 3	R/W	0000 0000$_H$	√	√	√	√	FFFE EC8C$_H$

(1) IPIR_CHn — Inter-PE Interrupt Register n (n = 0 to 3)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	PE3	—	PE1
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R/W

Table 3.61 IPIR_CHn Interrupt Register Contents

Bit Position	Bit Name	Function
2	PE3	Inter-PE Interrupt Request to PE3 Writing 1 to this bit makes an interrupt request to PE3. This bit is automatically cleared to 0 when the interrupt request has been notified. 0: Inter-PE interrupt request output is not specified or an interrupt request is not being output. 1: Interrupt request output is specified or an interrupt request is being output.
0	PE1	Inter-PE Interrupt Request to PE1 Writing 1 to this bit makes an interrupt request to PE1. This bit is automatically cleared to 0 when the interrupt request has been notified. 0: Inter-PE interrupt request output is not specified or an interrupt request is not being output. 1: Interrupt request output is specified or an interrupt request is being output.

3.2.4 Reliability Functions

3.2.4.1 PE Guard Function (PEG)

(1) Overview of the PEG Function

The PEG is a constituent of the slave guard system to prevent unauthorized access to the resources in the PE from the external master. This function protects access to the local RAM in the PE. In the initial state after a reset, all access by masters other than the PE itself is disabled. Setting the registers listed in **(3) List of PEG Protection Setting Registers**, enables access by masters other than the PE itself.

(1) Detecting PE guard violation

If the external master outside the PE makes an unauthorized access to the resource area in the PE for which PE guard is set, the access is detected as a PE guard violation.

(2) Blocking unauthorized accesses

When a PE guard violation is detected, unauthorized accesses to the internal resources of the PE are blocked to prevent the contents of PE resources from being modified illegally.

(3) Notifying occurrence of violation

When a PE guard violation is detected, it is notified to ECM. When the DMAC or DTS makes an unauthorized access, a DMA transfer error is detected.

(2) Protection Made by SPID

- Setting PEG Protection
  - Up to four areas can be set depending on the local RAM address of the own PE.
  - The area range is specified by the base address and the mask bit (4 Kbytes to 4 Gbytes).
  - “Read enable” and “write enable” can be set for each area.
  - “Enable” or “disable” can be selected on each system protection identifier (SPID) basis for each area.
- Access permission by the system protection identifier (SPID) (see **Figure 3.4**)

1. When the local RAM area is to be accessed, go to step 2.
   Otherwise, return an error response.
2. When any of enabled area 0 to area 3 is to be accessed, go to step 3.
   Otherwise, return an error response.
3. Whether or not all the conditions below for the relevant area are met.
   - The system protection identifier (SPID) is enabled.
   - Required operations (read/write) are enabled.
     Otherwise, return an error response.

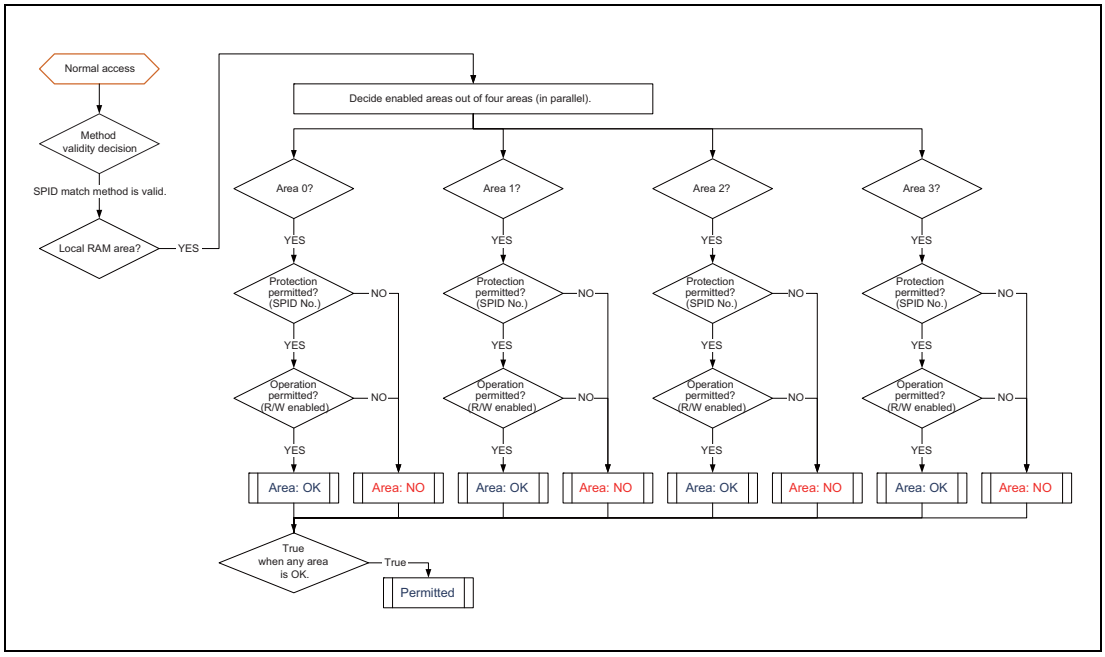


Figure 3.4 Access Permission by the System Protection Identifier (SPID)

(3) List of PEG Protection Setting Registers

Make necessary settings for the following registers to protect PE resources from unauthorized accesses by the external master.

- Accesses to the local RAM in the PE are permitted as detection targets.
- For accesses to the register set, no access restriction is provided independently for the PEG function. Set access protections such as IPG at the user side as necessary.
- PEG protection can be set by following the procedure below.

  1. Set the PE guard area n mask setting register as PEGGnMK.
  2. Set the PE guard area n base setting register (n = 0 to 3) as PEGGnBA.

Table 3.62 Base Address of PEG Register: FFFE_E600$_H$

Address Offset	Size (Byte)	Register Name	Abbreviation	Right	R/W	Operable Bit				Value after Reset
						1	8	16	32	
+00C$_H$	2	PE guard SPID master decision control register	PEGSP	—	R/W	—	√	√	—	0000$_H$
+080$_H$	4	PE guard area 0 mask setting register	PEGG0MK	—	R/W	—	√	√	√	0000 0000$_H$
+084$_H$	4	PE guard area 0 base setting register	PEGG0BA	—	R/W	—	√	√	√	0000 0000$_H$
+090$_H$	4	PE guard area 1 mask setting register	PEGG1MK	—	R/W	—	√	√	√	0000 0000$_H$
+094$_H$	4	PE guard area 1 base setting register	PEGG1BA	—	R/W	—	√	√	√	0000 0000$_H$
+0A0$_H$	4	PE guard area 2 mask setting register	PEGG2MK	—	R/W	—	√	√	√	0000 0000$_H$
+0A4$_H$	4	PE guard area 2 base setting register	PEGG2BA	—	R/W	—	√	√	√	0000 0000$_H$
+0B0$_H$	4	PE guard area 3 mask setting register	PEGG3MK	—	R/W	—	√	√	√	0000 0000$_H$
+0B4$_H$	4	PE guard area 3 base setting register	PEGG3BA	—	R/W	—	√	√	√	0000 0000$_H$

(4) Register Set

(a) PEGSP — PE Guard SPID Master Decision Control Register

This register is used to enable or disable access by an external master to the resources in the PE. The initial value of the SPEN bit is 0, which disables access to PE resources by an external master. Setting the SPEN bit to 1 enables access by an external master under the conditions set by PEGGnBA and PEGGnMK.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SPEN
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 3.63 PEGSP Register Contents

Bit Position	Bit Name	Function
15 to 1	—	Reserved These bits are always read as 0. The write value should always be 0.
0	SPEN	This bit enables or disables detection of accesses by the external master having SPID. 0: Detection of accesses by the external master having SPID is disabled. 1: Detection of accesses by the external master having SPID is enabled.

(b) PEGGnMK — PE Guard Area n Mask Setting Register

In combination with the PEGGnBA register, this register specifies a range or ranges within PE guard protection area n. Setting a GnMASK bit to 1 masks the corresponding address bit in the PEGGnBA register and places the corresponding area or areas inside the range of PE guard area n. The minimum size of the PE guard protection area is 4 KB.

Ex.) With the settings of PEGGnBA[31:12] = FEBF6$_H$ and PEGGnMK[31:12] = 00008$_H$, the PE guard protection area n is specified within the ranges from FEBF 6000$_H$ to FEBF 6FFF$_H$ and FEBF E000$_H$ to FEBF EFFF$_H$.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GnMASK															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GnMASK				—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R

Table 3.64 PEGGnMK Register Contents

Bit Position	Bit Name	Function
31 to 12	GnMASK	These bits determine whether to mask the base address PEGGnBA[31:12] that specifies the range of PE guard protection area n. 0: Target address bits are not compared when determining the PE guard area. 1: Target address bits are compared when determining the PE guard area.
11 to 0	—	Reserved These bits are always read as 0. The write value should always be 0.

(c) PEGGnBA — PE Guard Area n Base Setting Register

In combination with the PEGGnMK register, this register specifies a range or ranges within PE guard protection area n and sets the access enable conditions for the specified area. Setting the GnEN bit to 1 brings the address enable conditions specified by this register and the PEGGnMK register into effect.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GnBASE															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GnBASE				—	—	—	—	GnSP3	GnSP2	GnSP1	GnSP0	—	GnWR	GnRD	GnEN
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W

Table 3.65 PEGGnBA Register Contents

Bit Position	Bit Name	Function
31 to 12	GnBASE	Base address that specifies PE guard protection area n
11 to 8	—	Reserved
7	GnSP3	Enables accesses to PE guard protection area n by the external master. 0: Access by an external master having SPID = 3 is disabled. 1: Access by an external master having SPID = 3 is enabled.
6	GnSP2	Enables accesses to PE guard protection area n by the external master. 0: Access by an external master having SPID = 2 is disabled. 1: Access by an external master having SPID = 2 is enabled.
5	GnSP1	Enables accesses to PE guard protection area n by the external master. 0: Access by an external master having SPID = 1 is disabled. 1: Access by an external master having SPID = 1 is enabled.
4	GnSP0	Enables accesses to PE guard protection area n by the external master. 0: Access by an external master having SPID = 0 is disabled. 1: Access by an external master having SPID = 0 is enabled.
3	—	Reserved This bit is always read as 0. The write value should always be 0.
2	GnWR	Enables write access to PE guard protection area n. 0: Write access is disabled. 1: Write access is enabled.
1	GnRD	Enables read access to PE guard protection area n. 0: Read access is disabled. 1: Read access is enabled.
0	GnEN	PE guard protection area n enable 0: Settings for access enable conditions are disabled 1: Settings for access enable conditions are enabled

CAUTION

PEGGnBA.GnEN is cleared by writing to the PEGGnMK register.

3.2.4.2 PE's Internal Peripheral Device Protection Function (IPG)

(1) Overview of the IPG Function

The IPG is a system to prevent unauthorized accesses to peripheral devices from the CPU core equipped with the IPG. The IPG achieves the following functions. The IPG covers accesses to resources except the ROM and the local RAM.

(1) Detecting violation of peripheral device protection

If the CPU makes an unauthorized access to an area (peripheral device) for which peripheral device protection is set, the access is detected as "violation of peripheral device protection."

(2) Storing unauthorized-access information

When a violation of peripheral device protection is detected, the unauthorized-access information is stored in the IPG's internal register.

(3) Blocking unauthorized accesses

When a violation of peripheral device protection is detected, unauthorized accesses to peripheral devices are blocked to prevent contents of peripheral devices from being modified illegally.

(4) Notifying violation

When a violation of peripheral device protection is detected, a system error exception (SYSERR exception) request for generating an exception is made to ask the CPU to stop the processing.

For system error exceptions (SYSERR exceptions), see **Section 3.2.4.3, System Error Notification Control Function (SEG)**.

(5) Invalidating subsequent accesses

When a violation of peripheral device protection is detected, subsequent accesses (regardless of authorized or unauthorized accesses) are blocked until instructions from the CPU are received.

NOTE

Even if a request for generating an exception is immediately sent to the CPU in step (4) above, a subsequent access issued (before receiving a request from the IPG) by the CPU that does not know an occurrence of violation may illegally modify contents of peripheral devices. (Accesses after a violation has occurred result in unauthorized accesses.)

(2) IPG Function

(1) This function invalidates accesses according to their attributes (including address, transfer type, and access right).

(2) After an access right violation is detected until the error flag (described later) is cleared by writing by the software, subsequent accesses are invalidated. However, invalidation is applied only to accesses from the CPU and is not applied to accesses from outside the CPU core. Invalidation is performed independently of addresses.

(3) When a request for accessing different peripheral devices simultaneously is made due to misalignment or double-word access, the access is executed when all such accesses are enabled.

(3) IPG Protection Setting Registers for Illegal Users

To protect peripheral devices from unauthorized accesses by programs in user mode, the registers below require the settings listed in the table.

- Accesses in user mode are to be detected.
- This register set is intended for IPG settings related to user mode and reading of the IPG settings within its own machine.

Table 3.66 Base Address of IPG Register: FFFE E000$_H$

Address Offset	Size (Byte)	Register Name	Abbreviation	Right *1	R/W	Operable Bit				Value after Reset
						1	8	16	32	
+002$_H$	2	Peripheral device protection violation access information register	IPGECRUM	SV	R/W	—	—	√	—	Undefined (retained)
+008$_H$	4	Peripheral device protection violation access address register	IPGADRUM	SV	R/W	—	—	—	√	Undefined (retained)
+00D$_H$	1	Peripheral device protection enable register	IPGENUM	SV	R/W	√	√	—	—	00$_H$
+020$_H$	1	Peripheral device protection setting register 0	IPGPMTUM0	SV	R/W	√	√	—	—	00$_H$
+021$_H$	1	Peripheral device protection setting register 1	IPGPMTUM1	SV	R/W	√	√	—	—	00$_H$
+022$_H$	1	Peripheral device protection setting register 2	IPGPMTUM2	SV	R/W	√	√	—	—	00$_H$
+023$_H$	1	Peripheral device protection setting register 3	IPGPMTUM3	SV	R/W	√	√	—	—	00$_H$
+024$_H$	1	Peripheral device protection setting register 4	IPGPMTUM4	SV	R/W	√	√	—	—	00$_H$

Note 1. Registers for which “SV” is described are accessible by SV right (UM = 0).

(4) Register Set

(a) IPGECRUM — Peripheral Device Protection Violation Access Information Register

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	DS				EX	WR	RD	VD
Value after reset	0	0	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

×: Undefined (retained)

Table 3.67 Register Contents of IPGECRUM

Bit Position	Bit Name	Function
15, 14	—	Reserved These bits are always read as 0. The write value should always be 0.
13 to 8	—	Reserved These bits are read as an undefined value. The write value should always be 0.
7 to 4	DS	These bits store the data size of access that made a violation. 1000: Double-word (8 bytes) 0100: Word (4 bytes) 0010: Half-word (2 bytes) 0001: Byte Other than above: RFU
3	EX	This bit is set to 1 when a violation occurred in an instruction fetch read access. In other cases, this bit is cleared to 0.
2	WR	This bit is set to 1 when a violation occurred in a write access or bit operation or CAXI. In other cases, this bit is cleared to 0.
1	RD	This bit is set to 1 when a violation occurred in a read access or bit operation or CAXI. In other cases, this bit is cleared to 0.
0	VD	This bit is set to 1 when a violation of peripheral device protection is detected by a program with the relevant right. Even if another violation of peripheral device protection is detected while this bit is 1, data of this IPGECR register and the IPGADR register is not updated and is retained.

NOTE

When the IRE bit value of the IPGENUM register (described later) is 0 and violation of peripheral device protection by a program operating in user mode is an instruction fetch read access, no bit of this register is updated.

(b) IPGADRUM — Peripheral Device Protection Violation Access Address Register

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	EADR															
Value after reset	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	EADR															
Value after reset	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

×: Undefined (retained)

Table 3.68 Register Contents of IPGADRUM

Bit Position	Bit Name	Function
31 to 0	EADR	These bits store the address of the access in which a violation occurred.

NOTE

When the IRE bit value of the IPGENUM register (described later) is 0 and violation of peripheral device protection by a program operating in user mode is an instruction fetch read access, no bit of this register is updated.

(c) IPGENUM — Peripheral Device Protection Enable Register

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	IRE	E
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 3.69 Register Contents of IPGENUM

Bit Position	Bit Name	Function
7 to 2	—	Reserved These bits are always read as 0. The write value should always be 0.
1	IRE	This bit sets whether to store the access information in the peripheral device protection violation access address register and the peripheral device protection violation access information register when a violation of peripheral device protection occurred in an instruction fetch access. 0: Instruction fetch access information is not stored. 1: Instruction fetch access information is stored. **CAUTION** If you do not want to detect speculative instruction fetches (no instruction is executed in some cases), clear this bit to 0.
0	E	This bit enables or disables the peripheral devices protection function against accesses by the relevant access right. 0: The peripheral device protection function is disabled. 1: The peripheral device protection function is enabled.

(d) IPGPMTUM0 — Peripheral Device Protection Setting Register 0

Bit	7	6	5	4	3	2	1	0
	—	X1	W1	R1	—	X0	W0	R0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Table 3.70 Register Contents of IPGPMTUM0

Bit Position	Bit Name	Function
7	—	Reserved This bit is always read as 0. The write value should always be 0.
6	X1	This bit sets whether to enable instruction fetch read access to Peripheral Groups 0 to 3 and 5. 0: Instruction fetch read access to Peripheral Groups 0 to 3 and 5 is treated as violation. 1: Instruction fetch read access to Peripheral Groups 0 to 3 and 5 is not restricted.
5	W1	This bit sets whether to enable write access to Peripheral Groups 0 to 3 and 5. 0: Write access to Peripheral Groups 0 to 3 and 5 is treated as violation. 1: Write access to Peripheral Groups 0 to 3 and 5 is not restricted.
4	R1	This bit sets whether to enable read access to Peripheral Groups 0 to 3 and 5. 0: Read access to Peripheral Groups 0 to 3 and 5 is treated as violation. 1: Read access to Peripheral Groups 0 to 3 and 5 is not restricted.
3	—	Reserved This bit is always read as 0. The write value should always be 0.
2	X0	This bit sets whether to enable instruction fetch read access to peripheral devices to be connected to the H-Bus. 0: Instruction fetch read access to peripheral devices to be connected to the H-Bus is treated as violation. 1: Instruction fetch read access to peripheral devices to be connected to the H-Bus is not restricted.
1	W0	This bit sets whether to enable write access to peripheral devices to be connected to the H-Bus. 0: Write access to peripheral devices to be connected to the H-Bus is treated as violation. 1: Write access to peripheral devices to be connected to the H-Bus is not restricted.
0	R0	This bit sets whether to enable read access to peripheral devices to be connected to the H-Bus. 0: Read access to peripheral devices to be connected to the H-Bus is treated as violation. 1: Read access to peripheral devices to be connected to the H-Bus is not restricted.

(e) IPGPMTUM1 — Peripheral Device Protection Setting Register 1

Bit	7	6	5	4	3	2	1	0
	—	X1	—	—	—	X0	—	—
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R/W	R	R	R	R/W	R	R

Table 3.71 Register Contents of IPGPMTUM1

Bit Position	Bit Name	Function
7	—	Reserved This bit is always read as 0. The write value should always be 0.
6	X1	This bit sets whether to enable instruction fetch read access to GRAM Bank#1. 0: Instruction fetch read access to GRAM Bank#1 is treated as violation. 1: Instruction fetch read access to GRAM Bank#1 is not restricted.
5 to 3	—	Reserved These bits are always read as 0. The write value should always be 0.
2	X0	This bit sets whether to enable instruction fetch read access to GRAM Bank#0. 0: Instruction fetch read access to GRAM Bank#0 is treated as violation. 1: Instruction fetch read access to GRAM Bank#0 is not restricted.
1, 0	—	Reserved These bits are always read as 0. The write value should always be 0.

(f) IPGPMTUM2 — Peripheral Device Protection Setting Register 2

Bit	7	6	5	4	3	2	1	0
	—	—	W1	R1	—	—	W0	R0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R	R	R/W	R/W

Table 3.72 Register Contents of IPGPMTUM2

Bit Position	Bit Name	Function
7, 6	—	Reserved These bits are always read as 0. The write value should always be 0.
5	W1	This bit sets whether to enable write access to IPIR, MEV, and COMPTEST. 0: Write access to IPIR, MEV, and COMPTEST is treated as violation. 1: Write access to IPIR, MEV, and COMPTEST is not restricted.
4	R1	This bit sets whether to enable read access to IPIR, MEV, and COMPTEST. 0: Read access to IPIR, MEV, and COMPTEST is treated as violation. 1: Read access to IPIR, MEV, and COMPTEST is not restricted.
3, 2	—	Reserved These bits are always read as 0. The write value should always be 0.
1	W0	This bit sets whether to enable write access to INTC1. 0: Write access to INTC1 is treated as violation. 1: Write access to INTC1 is not restricted.
0	R0	This bit sets whether to enable read access to INTC1. 0: Read access to INTC1 is treated as violation. 1: Read access to INTC1 is not restricted.

(g) IPGPMTUM3 — Peripheral Device Protection Setting Register 3

Bit	7	6	5	4	3	2	1	0
	—	—	W1	R1	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R	R	R	R

Table 3.73 Register Contents of IPGPMTUM3

Bit Position	Bit Name	Function
7, 6	—	Reserved These bits are always read as 0. The write value should always be 0.
5	W1	This bit sets whether to enable write access to SysErrGen. 0: Write access to SysErrGen is treated as violation. 1: Write access to SysErrGen is not restricted.
4	R1	This bit sets whether to enable read access to SysErrGen. 0: Read access to SysErrGen is treated as violation. 1: Read access to SysErrGen is not restricted
3 to 0	—	Reserved These bits are always read as 0. The write value should always be 0.

(h) IPGPMTUM4 — Peripheral Device Protection Setting Register 4

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	W0	R0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 3.74 Register Contents of IPGPMTUM4

Bit Position	Bit Name	Function
7 to 2	—	Reserved These bits are always read as 0. The write value should always be 0.
1	W0	This bit sets whether to enable write access to the PEG. 0: Write access to the PEG is treated as violation. 1: Write access to the PEG is not restricted.
0	R0	This bit sets whether to enable read access the PEG. 0: Read access to the PEG is treated as violation. 1: Read access to the PEG is not restricted

3.2.4.3 System Error Notification Control Function (SEG)

Errors due to an instruction fetch or data access can be the sources of system error exceptions. A system error exception is an FE level exception from which return or recovery is not possible.

For source codes (FEIC) of the system error exceptions and error handling, see **Table 3.77, Error Factor Codes and Handling of G3M Core System Error Exceptions**.

SEG (SysErrGen) controls the notification and recording of the error by data access.

Errors due to instruction-fetch access are not conveyed via the SEG block, but the SEG block is notified of errors in RAM that has been cached in the instruction cache. For details, see **(2) Register Set**, **(a) SEGCONT — Error Notification Control Register**, and **(3) SEG Function**, **(c) Supplementary information on SYSERR exception**.

Multiple error occurrence inputs are categorized according to error factors, and are processed sequentially from the highest-priority error factor, generating an FE-level asynchronous exception (SYSERR).

The bit position of the SEGFLAG register becomes the priority of error factors. Error factors of lower bits take precedence over error factors of upper bits.

Error information is recorded once regardless of error frequency.

The error with the highest priority of error factor (in case errors occurred simultaneously) is valid. Recorded error information is not overwritten by subsequent errors.

(1) List of SEG Function Control Registers

Table 3.75 Base Address of SEG Register: FFFE E980$_H$

Address Offset	Size (Byte)	Register Name	Abbreviation	Right	R/W	Operable Bit				Value after Reset
						1	8	16	32	
+00$_H$	2	Error notification control register	SEGCONT	—	R/W *1	—	—	√	—	C774$_H$
+02$_H$	2	Error occurrence retention register	SEGFLAG	—	R/W *1	—	—	√	—	0000$_H$
+08$_H$	4	Error factor retention register (address)	SEGADDR	—	R/W *1	—	—	√	√	Undefined (retained)

Note 1. Write accesses from user mode are ignored.

NOTES

- If an access is made with an address offset or operable bits other than those specified above, an error response is returned.
- Write access is only possible with the supervisor mode (UM = 0). Attempting to write, if these conditions do not hold, leads to an error response being returned.
- No restriction is provided for read accesses.
  – Read accesses to ranges permitted by other protection systems are enabled at any time.

(2) Register Set

(a) SEGCONT — Error Notification Control Register

- This register is used to enable (= 1) or disable (= 0) notification of SYSERR request in response to error flags that store error occurrence status according to factors.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SS1E	—	—	—	—	—	VPGE	VCRE	—	TCME	ROME	VCIE	—	ICCE	—	—
Value after reset	1	1	0	0	0	1	1	1	0	1	1	1	0	1	0	0
R/W	R/W	R	R	R	R	R	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R	R

Table 3.76 SEGCONT Register Contents (1/2)

Bit Position	Bit Name	Function
15	SS1E	Notification of an address parity error in access to data in the local RAM for the given processor.
14	—	Reserved This bit is always read as 1. The write value should always be 1
13 to 11	—	Reserved These bits are always read as 0. The write value should always be 0.
10	—	Reserved This bit is always read as 1. The write value should always be 1.
9	VPGE	Notification of a response to an error in the P-Bus. The error includes the followings: • P-Bus guard error in writing access (P-bus guarding also applies to the guarding of all registers in the INTC2 module, and to all registers for peripheral DMA and the GRAM)
8	VCRE	Notification of the detection of illegal access by the IPG and subsequent access blocking.*2
7	—	Reserved This bit is always read as 0. The write value should always be 0.
6	TCME	Notification of an error in access to data in the local ROM for the given processor. The error includes the following cases: • When an ECC error which cannot be corrected occurs • When an access to the RAM-unimplemented area in the local RAM is detected
5	ROME	Notification of an error in access to the code flash when a table reference is read in response to a table reference interrupt. The error includes the following cases: • When an ECC error which cannot be corrected occurs • When an address parity error occurs

Table 3.76 SEGCONT Register Contents (2/2)

Bit Position	Bit Name	Function
4	VCIE	• Notification of a response to an error in the P-Bus (excluding errors in writing to the P-Bus). The error includes the following cases: – When an unimplemented area (FFFF 7900_H to FFFF $7EFF_H$) is accessed – P-Bus guard error in writing access ((P-bus guarding also applies to the guarding of all registers in the INTC2 module, and to all registers for peripheral DMA and the GRAM) – P-Bus data parity error (for a peripheral function which the data parity is applied to) – An error when DTSRAM is read • Notification of a response to an error in the code flash The error includes the following cases: – When an ECC error which cannot be corrected occurs – When an address parity error occurs • Notification of a response to an error in GRAM The error includes the following cases: – When an address parity error occurs – When an access protection violation occurs – When an ECC error which cannot be corrected occurs • Notification of a response to an error in data flash The error includes the following cases: – The occurrence of ECC error that cannot be corrected*1 • Notification of the detection of access to an interconnect reserved area – FFFF 0000_H to FFFF $4FFF_H$ – FFFE 0000_H to FFFE $BFFF_H$ – FB00 0000_H to FE9F $FFFF_H$ – F300 0000_H to F8FF $FFFF_H$ • Notification of the detection of illegal access by the IPG and subsequent access blocking*2 • Notification of violation of access permission – Read or write access to an IPG protection setting register in user mode (PSW.UM = 1) – Write access to an SEG function control register in user mode (PSW.UM = 1)
3	—	Reserved This bit is always read as 0. The write value should always be 0.
2	ICCE	Instruction cache error notification enable The error occurred in the instruction cache is handled when the instruction cache system register, ICCTRL.ICHEMK, is set to 0 (the value after reset is 1): Regarding the error occurred in the instruction cache, see **Section 3.2.1.2, Register Set**, **(7) ICERR — Instruction cache error register**.
1, 0	—	Reserved These bits are always read as 0. The write value should always be 0.

Note 1. This is a case where an uncorrectable ECC error has occurred while ECC error notification is enabled.

Note 2. For the address where an error occurred, see the description of the IPGADRUM register in **Section 3.2.4.2, PE's Internal Peripheral Device Protection Function (IPG)**.

Table 3.77 Error Factor Codes and Handling of G3M Core System Error Exceptions

Factor Codes	Error Contents
10	Reserved
11	Instruction fetch errors (from code flash memory)
12	Errors in which their notification is enabled by the SEGCONT second bit
13	Instruction fetch errors (from other than code flash memory)
14	Errors in which their notification is enabled by the SEGCONT fourth bit
15	Errors in which their notification is enabled by the SEGCONT fifth bit
16	Errors in which their notification is enabled by the SEGCONT sixth bit
17	Reserved
18	Errors in which their notification is enabled by the SEGCONT eighth bit
19	Errors in which their notification is enabled by the SEGCONT ninth bit
1A	Reserved
1B	Reserved
1C	Reserved
1D	Reserved
1E	Reserved
1F	Errors in which their notification is enabled by the SEGCONT fifteenth bit

(b) SEGFLAG — Error Occurrence Retention Register

- This register sets the flags of error occurrence, that store each error occurrence status by factors. Those are not automatically cleared (return to 0).
- Writing to the register enables both setting and clearing.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SS1F	—	—	—	—	—	VPGF	VCRF	—	TCMF	ROMF	VCIF	—	ICCF	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R	R	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R	R

Table 3.78 SEGFLAG Register Contents

Bit Position	Bit Name	Function
15	SS1F	Flag corresponding to bit 15 of the SEGCONT register
14 to 10	—	Reserved These bits are always read as 0. The write value should always be 0.
9	VPGF	Flag corresponding to bit 9 of the SEGCONT register
8	VCRF	Flag corresponding to bit 8 of the SEGCONT register
7	—	Reserved This bit is always read as 0. The write value should always be 0.
6	TCMF	Flag corresponding to bit 6 of the SEGCONT register
5	ROMF	Flag corresponding to bit 5 of the SEGCONT register
4	VCIF	Flag corresponding to bit 4 of the SEGCONT register
3	—	Reserved This bit is always read as 0. The write value should always be 0.
2	ICCF	Flag corresponding to bit 2 of the SEGCONT register
1, 0	—	Reserved These bits are always read as 0. The write value should always be 0.

(c) SEGADDR — Error Factor Retention Register (Address)

This register records information of an error factor that notifies a SYSERR request (only one history is recorded). It records the addresses where the error factors for the VCIF, ROMF, and TCMF bits of the SEGFLAG register were found. For the error factors which are not recorded, 0000 0000_H is stored. The value cannot be modified while the error occurrence flag to enable notification is set.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	Address[31:16]															
Value after reset	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	Address[15:0]															
Value after reset	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
R/W	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1

×: Undefined (retained)

Note 1. This cannot be modified when the error occurrence flag to enable notification is set.

Table 3.79 SEGADDR Register Contents

Bit Position	Bit Name	Function
31 to 0	Address[31:0]	These bits retain the address at which SYSERR source occurred. (If an error occurred by accessing to the local RAM area, the addresses of the 17 lower-order bits are retained and the addresses of the 15 higher-order bits are cleared to 0.)

(3) SEG Function

(a) SEG function: Notifying a SYSERR request due to an error flag

- Setting an error flag takes precedence over clearing the same flag.
  - Simultaneous clearing operation is ignored.
- Priority of error factors
  - The bit position of the notification-enabled SEGFLAG register becomes the priority of error factors. Error factors of lower bits take precedence over error factors of upper bits. Notification is made from the highest-priority error factor.
  - The bit position of error factors is notified as a “SYSERR factor code.”
- Conditions for starting SYSERR request notification
  - Even if a notification-disabled flag is set to 1, notification is not made.
  - Notification is made immediately after a notification-enabled flag is set to 1.
  - After clearing, notification is made depending on the flag state (re-arbitration).
- Finishing notification at a SYSERR request response
  - Even after notification is finished, the flag is not cleared automatically.
  - Notification is not made until re-arbitration is performed by setting or clearing the flag.
  - If an error flag that is prioritized higher than the error factor is set prior to a request response, the notification information may be replaced with an upper SYSERR factor code.

(b) SEG function: Recording error factor information

- When notification-enabled error occurrence is input, the error address is retained in the above register.
  - No information is retained by setting or clearing an error flag described in **(a) SEG function: Notifying a SYSERR request due to an error flag** above.
  - When multiple error occurrence inputs are present simultaneously, information other than the prioritized error factor is not retained.
- While the notification-enabled error flag described in **(a) SEG function: Notifying a SYSERR request due to an error flag** above is set to 1, overwrite to the above register is inhibited
  - If error occurrence input continues, information of subsequent error factors is not retained.
  - To reset the inhibition of overwrite to the register, clear either SEGCONT or SEGFLAG register (or both of them).

(c) Supplementary information on SYSERR exception

- The PSW.EBV bit retains its value and the base address for the exception handler is not changed even when a SYSERR exception is generated.
- Detection of errors in the instruction cache
  Even when an error is detected in the instruction cache, a rerun-typed SYSERR exception caused by instruction fetch does not occur. Execution of the instruction by the CPU still proceeds; the instruction cache automatically invalidates the entry which lead to the error and fetching is repeated, this time from the code flash memory. The SEG block will be informed that a cache error has occurred if the setting of the ICHEMK bit of the ICCTRL cache-control system register is 0. See **(7) ICERR — Instruction cache error register** in **Section 3.2.1.2, Register Set** for the details on instruction cache error.

3.2.4.4 Checker Core

The CPU1 has the checker core for safety assurance, resulting in a highly reliable system. Monitoring the outputs from the CPU1 and the checker core with the comparator all the time enables immediate detection of the CPU1 abnormal operations. Duplication by the checker core covers the CPU core, FPU, MPU, PEG, IPG, SEG, and INTC1. The CPU1 can also conduct a fault diagnosis test of its own comparator through a pseudo-error generated by the COMPTEST module. For details of the COMPTEST module, see **Section 29, Safety**.

CAUTION

Reading of any register with an undefined value after a reset in a PE or writing to memory or a register outside the PE may cause a lock step comparison error. The values of some program registers and system registers are undefined after a reset, so pay attention to this when saving register values on the stack in RAM.

3.3 PCU

3.3.1 Core Functions

3.3.1.1 Features

Table 3.80 lists features of the RH850G3K core.

Table 3.80 Features of the RH850G3K Core

Item	Feature
CPU	• Advanced 32-bit architecture for embedded control • 32-bit internal data bus • Thirty-two 32-bit general-purpose registers • RISC-type instruction sets – Long-/short-format load/store instructions – Three-operand instructions – Instruction sets based on C language • CPU operating modes – User mode and supervisor mode • Address space: 128-Mbyte linear address space for both data and instructions • Instructions: A snooze instruction (SNOOZE) is included for temporary suspension by switching the high-speed peripheral clock signal (CLK_HSB) off for 32 clock cycles.
Coprocessor	• Coprocessor is not incorporated
Exception/ Interrupt	• 8 interrupt priority levels settable for each channel • Vector selection method selectable according to performance request or memory usage – Direct branching exception vectors – Indirect branching exception vectors referring to the address table • Supports the high-speed save/restore processing of the context by the dedicated instructions (PUSHSP and POPSP) at the generation of an interrupt
Memory Management	• Memory protection function (MPU): four areas settable
Cache	• Cache is not incorporated

3.3.1.2 Register Set

(1) Program Registers

Program registers include the general-purpose registers (r0 to r31) and the program counter (PC).

The general-purpose register r0 always retains 0, but the values of the registers r1 to r31 after reset are undefined.

Table 3.81 Program Registers

Program Register	Name	Function	Description
General-purpose registers	r0	Zero register	Always retains “0”
	r1	Assembler reserved register	Used as a working register for generating addresses
	r2	Register for address and data variables (used when the real-time OS used does not use this register)	
	r3	Stack pointer (SP)	Used for generating a stack frame when a function is called
	r4	Global pointer (GP)	Used for accessing a global variable in the data area
	r5	Text pointer (TP)	Used as a register that indicates the start of the text area (area where program code is placed)
	r6 to r29	Register for addresses and data variables	
	r30	Element pointer (EP)	Used as a base pointer for generating addresses when accessing memory
	r31	Link pointer (LP)	Used when the compiler calls a function
Program counter	PC	Retains instruction addresses during execution of programs	

NOTE

For further descriptions of r1, r3 to r5, and r31 used for an assembler and/or C compiler, see the specification of each software development environment.

(a) General-Purpose Registers

A total of 32 general-purpose registers (r0 to r31) are provided. All of these registers can be used for either data variables or address variables. Of the general-purpose registers, r0 to r5, r30, and r31 are assumed to be used for special purposes in software development environments, so it is necessary to note the following when using them.

1. r0, r3, and r30

   These registers are implicitly used by instructions.

   r0 is a register that always retains "0". It is used for operations that use 0, addressing with base address being 0, etc.

   r3 is implicitly used by the PREPARE, DISPOSE, PUSHSP, and POPSP instructions.

   r30 is used as a base pointer when the SLD instruction or SST instruction accesses memory.

2. r1, r4, r5, and r31

   These registers are implicitly used by the assembler and C compiler.

   When using these registers, register contents must first be saved so they are not lost and can be restored after the registers are used.

3. r2

   This register might be used by a real-time OS in some cases. If the real-time OS that is being used is not using r2, r2 can be used as a register for address variables or data variables.

(b) PC — Program Counter

The PC retains the address of the instruction being executed.

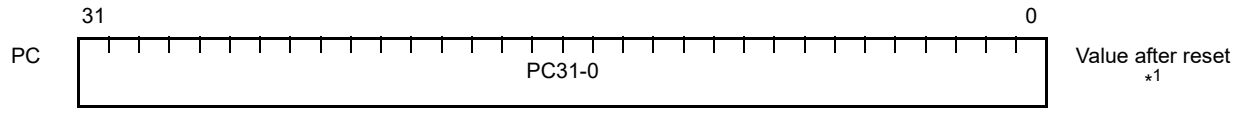


Table 3.82 PC Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 1	PC31-1	These bits indicate the address of the instruction being executed.	R/W	*1
0	PC0	This bit is fixed to 0, and branching to an odd number address is disabled.	R/W	0

Note 1. In this product, the value after reset differs depending on the startup area. For details, see **Section 4, Address Space**.

CAUTION

For a CPU whose instruction addressing range is 128 MB, a value resulting from a sign-extension of bit 26 is automatically set to bits 31 to 27.

(2) Basic System Registers

The basic system registers are used to control CPU status and to retain exception information.

System registers are read from or written to by using the LDSR or STSR instructions and specifying the system register number, which is made up of a register number and selection ID.

Table 3.83 Basic System Registers

Register No. (reg.ID, sel.ID)	Symbol	Function	Access Permission
SR0, 0	EIPC*1	Status save registers when acknowledging EI level exception	SV
SR1, 0	EIPSW	Status save registers when acknowledging EI level exception	SV
SR2, 0	FEPC*1	Status save registers when acknowledging FE level exception	SV
SR3, 0	FEPSW	Status save registers when acknowledging FE level exception	SV
SR 5, 0	PSW	Program status word	*2
SR 13, 0	EIIC	EI level exception cause	SV
SR 14, 0	FEIC	FE level exception cause	SV
SR 16, 0	CTPC *1	CALLT execution status save register	UM
SR 17, 0	CTPSW	CALLT execution status save register	UM
SR 20, 0	CTBP *1	CALLT base pointer	UM
SR 28, 0	EIWR	EI level exception working register	SV
SR 29, 0	FEWR	FE level exception working register	SV
SR 31, 0	(BSEL)	(Reserved for backwards compatibility with the V850E2 series)*3	SV
SR0, 1	MCFG0	Machine configuration	SV
SR2, 1	RBASE	Reset vector base address	SV
SR3, 1	EBASE*1	Exception handler vector address	SV
SR4, 1	INTBP*1	Base address of the interrupt handler table	SV
SR 5, 1	MCTL	CPU control	SV
SR 6, 1	PID	Processor ID	SV
SR 11, 1	SCCFG	SYSCALL operation setting	SV
SR 12, 1	SCBP*1	SYSCALL base pointer	SV
SR0, 2	HTCFG0	Process number	SV
SR6, 2	MEA*1	Memory error address	SV
SR7, 2	ASID	Address space ID	SV
SR8, 2	MEI	Memory error information	SV

Note 1. For a CPU whose instruction addressing range is 128 MB, a value resulting from a sign-extension of bit 26 is automatically set to bits 31 to 27.

Note 2. The access permission differs depending on the bit. For details, see **Table 3.88, Access Permission for PSW Register**.

Note 3. Reserved for backwards compatibility with the V850E2 series. These registers are always read as 0. Writing to these registers is ignored.

(a) EIPC — Status save register when acknowledging EI level exception register

When an EI level exception is acknowledged, the address of the instruction that was being executed when the EI level exception occurred, or of the next instruction, is saved to the EIPC register (see *section 4.1.3 Types of Exceptions* in *RH850G3K User's Manual: Software*) .

Because there is only one pair of EI level exception status save registers, when processing multiple exceptions, the contents of these registers must be saved by a program.

Be sure to set an even-numbered address to the EIPC register. An odd-numbered address must not be specified.

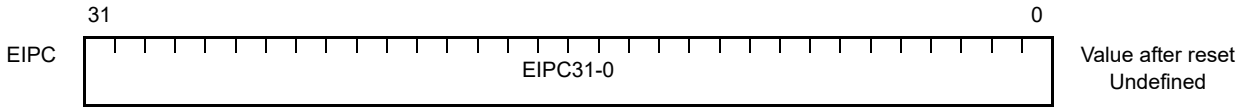


Table 3.84 EIPC Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 1	EIPC31-1	These bits indicate the PC saved when an EI level exception is acknowledged.	R/W	Undefined
0	EIPC0	This bit indicates the PC saved when an EI level exception is acknowledged. Always set this bit to 0. Even if it is set to 1, the value transferred to the PC when the EIRET instruction is executed is 0.	R/W	Undefined

CAUTION

For a CPU whose instruction addressing range is 128 MB, a value resulting from a sign-extension of bit 26 is automatically set to bits 31 to 27 of EIPC.

(b) EIPSW — Status save register when acknowledging EI level exception register

When an EI level exception is acknowledged, the current PSW setting is saved to the EIPSW register.

Because there is only one pair of EI level exception status save registers, when processing multiple exceptions, the contents of these registers must be saved by a program.

CAUTION

Bits 11 to 9 cannot usually be changed since these bits are associated with debugging.

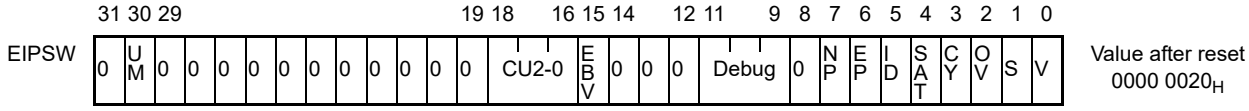


Table 3.85 EIPSW Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
30	UM	This bit stores the PSW.UM bit setting when an EI level exception is acknowledged.	R/W	0
29 to 19	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
18 to 16	CU2-0	These bits store the PSW.CU2 to 0 field setting when an EI level exception is acknowledged. (Reserved for future expansion. Be sure to clear to 0.)	R	0
15	EBV	This bit stores the PSW.EBV bit setting when an EI level exception is acknowledged.	R/W	0
14 to 12	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
11 to 9	Debug	These bits store the PSW.Debug field when an EI level exception is acknowledged.	R/W	0
8	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
7	NP	This bit stores the PSW.NP bit setting when an EI level exception is acknowledged.	R/W	0
6	EP	This bit stores the PSW.EP bit setting when an EI level exception is acknowledged.	R/W	0
5	ID	This bit stores the PSW.ID bit setting when an EI level exception is acknowledged.	R/W	1
4	SAT	This bit stores the PSW.SAT bit setting when an EI level exception is acknowledged.	R/W	0
3	CY	This bit stores the PSW.CY bit setting when an EI level exception is acknowledged.	R/W	0
2	OV	This bit stores the PSW.OV bit setting when an EI level exception is acknowledged.	R/W	0
1	S	This bit stores the PSW.S bit setting when an EI level exception is acknowledged.	R/W	0
0	Z	This bit stores the PSW.Z bit setting when an EI level exception is acknowledged.	R/W	0

(c) FEPC — Status save register when acknowledging FE level exception register

When an FE level exception is acknowledged, the address of the instruction that was being executed when the FE level exception occurred, or of the next instruction, is saved to the FEPC register (see *section 4.1.3 Types of Exceptions* in *RH850G3K User's Manual: Software*) in the Software Manual.

Because there is only one pair of FE level exception status save registers, when processing multiple exceptions, the contents of these registers must be saved by a program.

Be sure to set an even-numbered address to the FEPC register. An odd-numbered address must not be specified.

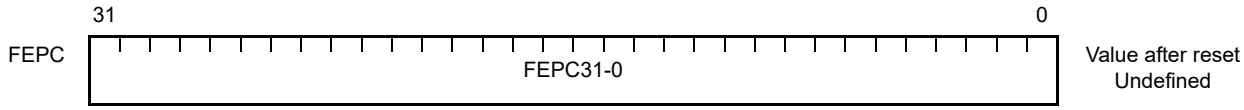


Table 3.86 FEPC Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 1	FEPC31-1	These bits indicate the PC saved when an FE level exception is acknowledged.	R/W	Undefined
0	FEPC0	This bit indicates the PC saved when an FE level exception is acknowledged. Always set this bit to 0. Even if it is set to 1, the value transferred to the PC when the FERET instruction is executed is 0.	R/W	Undefined

CAUTION

For a CPU whose instruction addressing range is 128 MB, a value resulting from a sign-extension of bit 26 is automatically set to bits 31 to 27 of FEPC.

(d) FEPSW — Status save register when acknowledging FE level exception register

When an FE level exception is acknowledged, the current PSW setting is saved to the FEPSW register.

Because there is only one pair of FE level exception status save registers, when processing multiple exceptions, the contents of these registers must be saved by a program.

CAUTION

Bits 11 to 9 cannot usually be changed since these bits are associated with debugging.

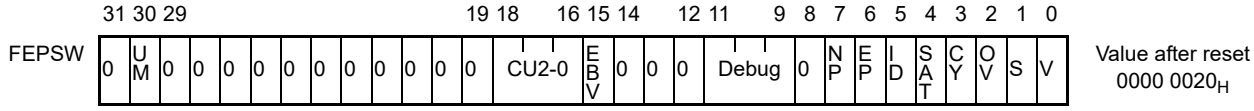


Table 3.87 FEPSW Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
30	UM	This bit stores the PSW.UM bit setting when an FE level exception is acknowledged.	R/W	0
29 to 19	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
18 to 16	CU2-0	These bits store the PSW.CU2 to 0 field setting when an FE level exception is acknowledged. (Reserved for future expansion. Be sure to clear to 0.)	R	0
15	EBV	This bit stores the PSW.EBV bit setting when an FE level exception is acknowledged.	R/W	0
14 to 12	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
11 to 9	Debug	These bits store the PSW.Debug field when an FE level exception is acknowledged.	R/W	0
8	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
7	NP	This bit stores the PSW.NP bit setting when an FE level exception is acknowledged.	R/W	0
6	EP	This bit stores the PSW.EP bit setting when an FE level exception is acknowledged.	R/W	0
5	ID	This bit stores the PSW.ID bit setting when an FE level exception is acknowledged.	R/W	1
4	SAT	This bit stores the PSW.SAT bit setting when an FE level exception is acknowledged.	R/W	0
3	CY	This bit stores the PSW.CY bit setting when an FE level exception is acknowledged.	R/W	0
2	OV	This bit stores the PSW.OV bit setting when an FE level exception is acknowledged.	R/W	0
1	S	This bit stores the PSW.S bit setting when an FE level exception is acknowledged.	R/W	0
0	Z	This bit stores the PSW.Z bit setting when an FE level exception is acknowledged.	R/W	0

(e) PSW — Program status word register

PSW (program status word) is a set of flags that indicate the program status (instruction execution result) and bits that indicate the operation status of the CPU (flags are bits in the PSW that are referenced by a condition instruction (Bcond, CMOV, etc.)).

When the LDSR instruction is used to change the contents of each bit in this register, the changed contents become valid from the instruction following the LDSR instruction.

The access permission for the PSW register differs depending on the bit. All bits can be read, but some bits can only be written under certain conditions. See **Table 3.88, Access Permission for PSW Register** for the access permission for each bit.

Table 3.88 Access Permission for PSW Register

Bit		Access Permission When Reading	Access Permission When Writing
30	UM	UM	SV*1
18 to 16	CU2-0	UM	SV*1
15	EBV	UM	SV*1
11 to 9	Debug	UM	Special*1
7	NP	UM	SV*1
6	EP	UM	SV*1
5	ID	UM	SV*1
4	SAT	UM	UM
3	CY	UM	UM
2	OV	UM	UM
1	S	UM	UM
0	Z	UM	UM

Note 1. The access permission for the whole PSW register is UM, so the PIE exception does not occur even if the register is written by using an LDSR instruction when PSW.UM is 1. In this case, writing is ignored.

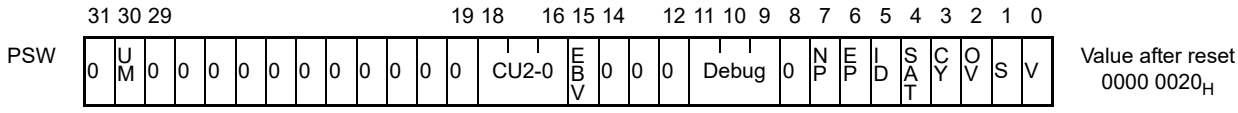


Table 3.89 PSW Register Contents (1/2)

Bit Position	Bit Name	Function	R/W	Value after Reset
31	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
30	UM	This bit indicates that the CPU is in user mode (UM mode) 0: Supervisor mode 1: User mode	R/W	0
29 to 19	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
18 to 16	CU2-0	(Reserved for future expansion. Be sure to clear to 0.)	R	000
15	EBV	This bit indicates the reset vector and exception vector operation. See the descriptions of the RBASE register in **(q) RBASE — Reset vector base address register** and the EBASE register in **(r) EBASE — Exception handler vector address register**.	R/W	0
14 to 12	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
11 to 9	Debug	These bits are used in the debugging functions of development tools. In normal operation, clear these bits to 0.	—	0
8	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0

Table 3.89 PSW Register Contents (2/2)

Bit Position	Bit Name	Function	R/W	Value after Reset
7	NP	Disables acceptance of FE level exceptions. Once an FE level exception is accepted, this bit is set to 1 and disables acceptance of EI level exceptions or FE level exceptions. For the exceptions disabled by the NP bit, see *table 4-1 Exception Cause List* in *RH850G3K User's Manual: Software*. 0 : An acceptance of FE level exceptions is enabled 1 : An acceptance of FE level exceptions is disabled	R/W	0
6	EP	Disables acceptance of EI level exceptions. Once an IE level exception or FE level exception is accepted, this bit is set to 1 and disables acceptance of EI level exceptions. For the exceptions disabled by the ID bit, see *table 4-1 Exception Cause List* in *RH850G3K User's Manual: Software*. This bit does not affect acknowledging an exception request even when it is set to 1. A change in the ID bit by the IE or DI instruction is enabled from the next instruction. 0: An interrupt is not being serviced. 1: An exception other than an interrupt is being serviced.	R/W	0
5	ID	Disables acceptance of EI level exceptions. Once an IE level exception or FE level exception is accepted, this bit is set to 1 and disables acceptance of EI level exceptions. For the exceptions disabled by the ID bit, see *table 4-1 Exception Cause List* in *RH850G3K User's Manual: Software*. This bit is also used to disable EI level exceptions from being acknowledged as a critical section while an ordinary program or interrupt is being serviced. It is set to 1 when the DI instruction is executed, and cleared to 0 when the EI instruction is executed. A change in the ID bit by the IE or DI instruction is enabled from the next instruction. 0 : An acceptance of IE level exceptions is enabled 1 : An acceptance of IE level exceptions is disabled	R/W	1
4	SAT*1	This bit indicates that the operation result is saturated because the result of a saturated operation instruction operation has overflowed. This is a cumulative flag, so when the operation result of the saturated operation instruction becomes saturated, this bit is set to 1, but it is not cleared to 0 when the operation result for a subsequent instruction is not saturated. This bit is cleared to 0 by the LDSR instruction. This bit is neither set to 1 nor cleared to 0 when an arithmetic operation instruction is executed. 0: Not saturated 1: Saturated	R/W	0
3	CY	This bit indicates whether a carry or borrow has occurred in the operation result. 0: Carry and borrow have not occurred. 1: Carry or borrow has occurred.	R/W	0
2	OV*1	This bit indicates whether or not an overflow has occurred during an operation. 0: Overflow has not occurred. 1: Overflow has occurred.	R/W	0
1	S*1	This bit indicates whether or not the result of an operation is negative. 0: Result of operation is positive or 0. 1: Result of operation is negative	R/W	0
0	Z	This bit indicates whether or not the result of an operation is 0. 0: Result of operation is not 0. 1: Result of operation is 0.	R/W	0

Note 1. The operation result of the saturation processing is determined in accordance with the contents of the OV flag and S flag during a saturated operation. When only the OV flag is set to 1 during a saturated operation, the SAT flag is set to 1.

Operation result status	Flag status			Operation result after saturation processing
	SAT	OV	S	
Exceeded positive maximum value	1	1	0	7FFF FFFF$_H$
Exceeded negative maximum value	1	1	1	8000 0000$_H$

Operation result status	Flag status			Operation result after saturation processing
	SAT	OV	S	
Positive (maximum value not exceeded)	Value prior to operation is retained.	0	0	Operation result itself
Negative (maximum value not exceeded)			1	

(f) EIIC — EI level exception cause register

The EIIC register retains the cause of any EI level exception that occurs. The value retained in this register is an exception code corresponding to a specific exception cause.

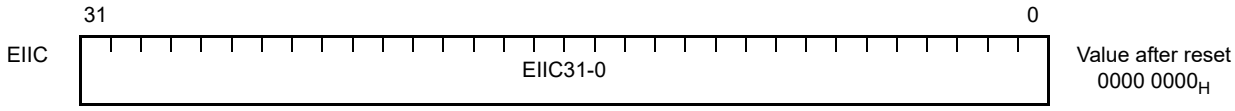


Table 3.90 EIIC Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 0	EIIC31-0	These bits store the exception cause code when an EI level exception occurs. Regarding the stored exception source code, see *Table 6.14, Interrupt Exception Handler and Priority* and *table 4-1 Exception Cause List* in *RH850G3K User's Manual: Software*. The EIIC31-16 field stores detailed exception cause codes defined individually for each exception. If there is no particular definition, these bits are set to 0.	R/W	0

(g) FEIC — FE level exception cause register

The FEIC register retains the cause of any FE level exception that occurs. The value retained in this register is an exception code corresponding to a specific exception cause.

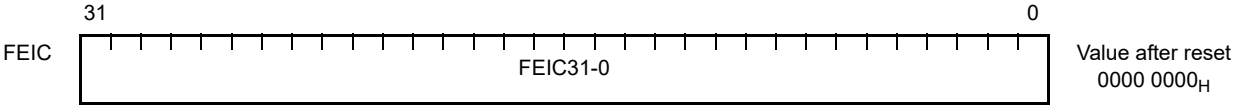


Table 3.91 FEIC Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 0	FEIC31-0	These bits store the exception cause code when an FE level exception occurs. Regarding the stored exception source code, see *Table 6.14, Interrupt Exception Handler and Priority* and *table 4-1 Exception Cause List* in *RH850G3K User's Manual: Software*. The FEIC31-16 field stores detailed exception cause codes defined individually for each exception. If there is no particular definition, these bits are set to 0.	R/W	0

(h) CTPC — Status save register when executing CALLT register

When a CALLT instruction is executed, the address of the next instruction after the CALLT instruction is saved to CTPC. Be sure to set an even-numbered address to the CTPC register. An odd-numbered address must not be specified.

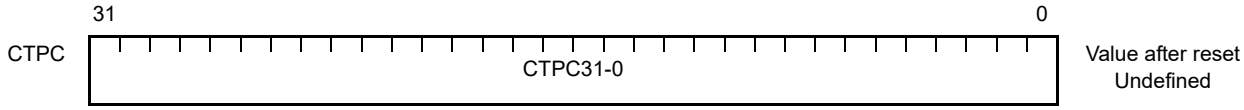


Table 3.92 CTPC Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 1	CTPC31-1	These bits indicate the PC of the instruction after the CALLT instruction.	R/W	Undefined
0	CTPC0	This bit indicates the PC of the instruction after the CALLT instruction. Always set this bit to 0. Even if it is set to 1, the value transferred to the PC when the CTRET instruction is executed is 0.	R/W	Undefined

CAUTION

For a CPU whose instruction addressing range is 128 MB, a value resulting from a sign-extension of bit 26 is automatically set to bits 31 to 27 of CTPC.

(i) CTPSW — Status save register when executing CALLT register

When a CALLT instruction is executed, some of the PSW (program status word) settings are saved to CTPSW.

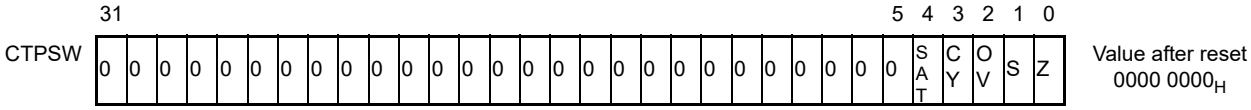


Table 3.93 CTPSW Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 5	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
4	SAT	This bit stores the PSW.SAT bit setting when the CALLT instruction is executed.	R/W	0
3	CY	This bit stores the PSW.CY bit setting when the CALLT instruction is executed.	R/W	0
2	OV	This bit stores the PSW.OV bit setting when the CALLT instruction is executed.	R/W	0
1	S	This bit stores the PSW.S bit setting when the CALLT instruction is executed.	R/W	0
0	Z	This bit stores the PSW.Z bit setting when the CALLT instruction is executed.	R/W	0

(j) CTBP — CALLT base pointer register

The CTBP register is used to specify table addresses of the CALLT instruction and generate target addresses. Be sure to set the CTBP register to a half word address.

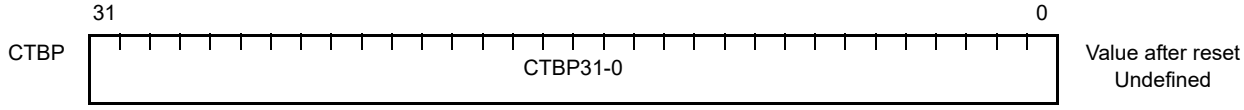


Table 3.94 CTBP Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 1	CTBP31-1	These bits indicate the base pointer address of the CALLT instruction. These bits indicate the first address in the table used by the CALLT instruction.	R/W	Undefined
0	CTBP0	These bits indicate the base pointer address of the CALLT instruction. These bits indicate the first address in the table used by the CALLT instruction. Always set this bit to 0.	R	0

CAUTION

For a CPU whose instruction addressing range is 128 MB, a value resulting from a sign-extension of bit 26 is automatically set to bits 31 to 27 of CTBP.

(k) ASID — Address space ID register

This is the address space ID. This is used to identify the address space provided by the memory management function.

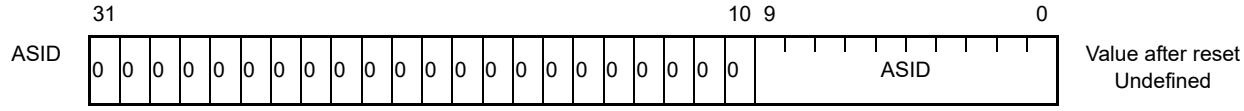


Table 3.95 ACID Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 10	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
9 to 0	ASID	This is the address space ID.	R/W	Undefined

(l) EIWR — EI level exception working register

The EIWR register is used as a working register when an EI level exception has occurred.

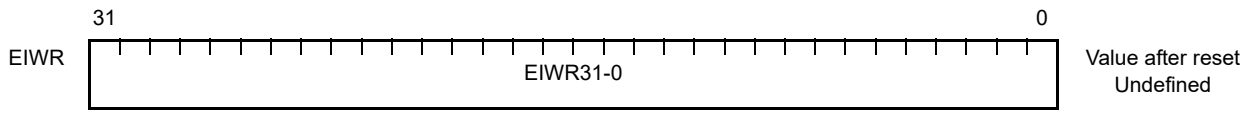


Table 3.96 EIWR Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 0	EIWR31-0	These bits constitute a working register that can be used for any purpose during the processing of an EI level exception. Use this register for purposes such as storing the values of general-purpose registers.	R/W	Undefined

(m) FEWR — FE level exception working register

The FEWR register is used as a working register when an FE level exception has occurred.

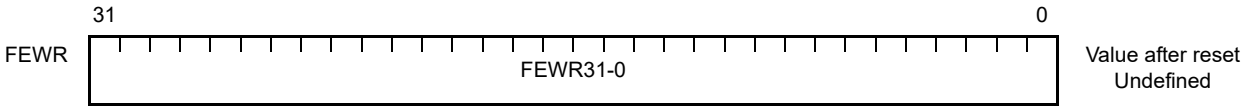


Table 3.97 FEWR Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 0	FEWR31-0	These bits constitute a working register that can be used for any purpose during the processing of an FE level exception. Use this register for purposes such as storing the values of general-purpose registers.	R/W	Undefined

(n) HTCFG0 — Thread configuration register

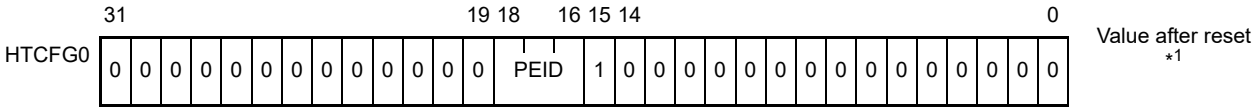


Table 3.98 HTCFG0 Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 19	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
18 to 16	PEID	These bits indicate the processor element number.	R	*2
15	—	(Reserved for future expansion. Be sure to set this bit to 1.)	R	1
14 to 0	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0

Note 1. The value is 0003 8000_H in PCU (PE3) of this product.
Note 2. The value is 011_B in PCU (PE3) of this product.

(o) MEA — Memory error address register

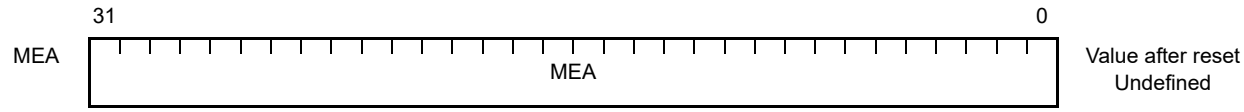


Table 3.99 MEA Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 0	MEA	These bits store the address when an MAE (misaligned) or MPU exception occurs.	R/W	Undefined

(p) MEI — Memory error information register

This register is used to store information about the instruction that caused the exception when a misaligned (MAE) or memory protection (MDP) exception occurs. This information is used during emulation.

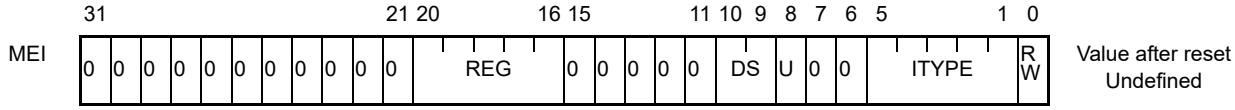


Table 3.100 MEI Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 21	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
20 to 16	REG4-0	These bits indicate the number of the source or destination register accessed by the instruction that caused the exception. For details, see **Table 3.101, Instructions Causing Exceptions and Values of MEI Register**.	R/W	Undefined
15 to 11	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
10, 9	DS	These bits indicate the type of data handled by the instruction that caused the exception.*1 0: Byte (8 bits) 1: Half word (16 bits) 2: Word (32 bits) 3: Double-word (64 bits) For details, see **Table 3.101, Instructions Causing Exceptions and Values of MEI Register**.	R/W	Undefined
8	U	These bits indicate the sign extension method of the instruction that caused the exception. 0: Signed 1: Unsigned For details, see **Table 3.101, Instructions Causing Exceptions and Values of MEI Register**.	R/W	Undefined
7, 6	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
5 to 1	ITYPE4-0	These bits indicate the instruction that caused the exception. For details, see **Table 3.101, Instructions Causing Exceptions and Values of MEI Register**.	R/W	Undefined
0	RW	This bit indicates whether the operation of the instruction that caused the exception was read (Load-memory) or write (Store-memory) 0: Read (Load-memory) 1: Write (Store-memory) For details, see **Table 3.101, Instructions Causing Exceptions and Values of MEI Register**.	R/W	Undefined

Note 1. Even if the data is divided and access is made several times due to the specifications of the hardware, the original data type indicated by the instruction is stored.

Table 3.101 Instructions Causing Exceptions and Values of MEI Register (1/2)

Instruction	REG	DS	U	RW	ITYPE
SLD.B	dst	0 (Byte)	0 (Signed)	0 (Read)	00000_B
SLD.BU	dst	0 (Byte)	1 (Unsigned)	0 (Read)	00000_B
SLD.H	dst	1 (Half word)	0 (Signed)	0 (Read)	00000_B
SLD.HU	dst	1 (Half word)	1 (Unsigned)	0 (Read)	00000_B
SLD.W	dst	2 (Word)	0 (Signed)	0 (Read)	00000_B
SST.B	src	0 (Byte)	0 (Signed)	1 (Write)	00000_B
SST.H	src	1 (Half word)	0 (Signed)	1 (Write)	00000_B

Table 3.101 Instructions Causing Exceptions and Values of MEI Register (2/2)

Instruction	REG	DS	U	RW	ITYPE
SST.W	src	2 (Word)	0 (Signed)	1 (Write)	00000_B
LD.B (disp16)	dst	0 (Byte)	0 (Signed)	0 (Read)	00001_B
LD.BU (disp16)	dst	0 (Byte)	1 (Unsigned)	0 (Read)	00001_B
LD.H (disp16)	dst	1 (Half word)	0 (Signed)	0 (Read)	00001_B
LD.HU (disp16)	dst	1 (Half word)	1 (Unsigned)	0 (Read)	00001_B
LD.W (disp16)	dst	2 (Word)	0 (Signed)	0 (Read)	00001_B
ST.B (disp16)	src	0 (Byte)	0 (Signed)	1 (Write)	00001_B
ST.H (disp16)	src	1 (Half word)	0 (Signed)	1 (Write)	00001_B
ST.W (disp16)	src	2 (Word)	0 (Signed)	1 (Write)	00001_B
LD.B (disp23)	dst	0 (Byte)	0 (Signed)	0 (Read)	00010_B
LD.BU (disp23)	dst	0 (Byte)	1 (Unsigned)	0 (Read)	00010_B
LD.H (disp23)	dst	1 (Half word)	0 (Signed)	0 (Read)	00010_B
LD.HU (disp23)	dst	1 (Half word)	1 (Unsigned)	0 (Read)	00010_B
LD.W (disp23)	dst	2 (Word)	0 (Signed)	0 (Read)	00010_B
ST.B (disp23)	src	0 (Byte)	0 (Signed)	1 (Write)	00010_B
ST.H (disp23)	src	1 (Half word)	0 (Signed)	1 (Write)	00010_B
ST.W (disp23)	src	2 (Word)	0 (Signed)	1 (Write)	00010_B
LD.DW (disp23)	dst	3 (Double-word)	0 (Signed)	0 (Read)	00010_B
ST.DW (disp23)	src	3 (Double-word)	0 (Signed)	1 (Write)	00010_B
LDL.W	dst	2 (Word)	0 (Signed)	0 (Read)	00111_B
STC.W	src	2 (Word)	0 (Signed)	1 (Write)	00111_B
CAXI	dst	2 (Word)	1 (Unsigned)	0 (Read)*1	01000_B
SET1	—	0 (Byte)	1 (Unsigned)	0 (Read)*1	01001_B
CLR1	—	0 (Byte)	1 (Unsigned)	0 (Read)*1	01001_B
NOT1	—	0 (Byte)	1 (Unsigned)	0 (Read)*1	01001_B
TST1	—	0 (Byte)	1 (Unsigned)	0 (Read)	01001_B
PREPARE	—	2 (Word)	1 (Unsigned)	1 (Write)	01100_B
DISPOSE	—	2 (Word)	1 (Unsigned)	0 (Read)	01100_B
PUSHSP	—	2 (Word)	1 (Unsigned)	1 (Write)	01101_B
POPSP	—	2 (Word)	1 (Unsigned)	0 (Read)	01101_B
SWITCH	—	1 (Half word)	0 (Signed)	0 (Read)	10000_B
CALLT	—	1 (Half word)	1 (Unsigned)	0 (Read)	10001_B
SYSCALL	—	2 (Word)	1 (Unsigned)	0 (Read)	10010_B
Interrupt (table reference)*2	—	2 (Word)	1 (Unsigned)	0 (Read)	10101_B

Note 1. This exception occurs when the instruction executes a read access.

Note 2. An exception occurs when the table reference interrupt vector is read.

NOTE

dst: destination register number, src: source register number

(q) RBASE — Reset vector base address register

This register indicates the reset vector address when there is a reset. If the PSW.EBV bit is 0, this vector address is also used as the exception vector address.

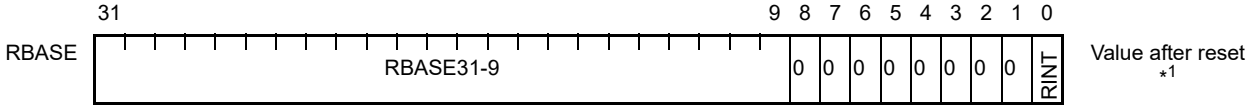


Table 3.102 RBASE Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 9	RBASE31-9	These bits indicate the reset vector when there is a reset. When PSW.EBV = 0, this address is also used as the exception vector. The EBASE8 to 0 bits are implicitly set to 0.	R	*1
8 to 1	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
0	RINT	When the RINT bit is set, the offset address of the exception handler used during interrupt servicing is decremented. This bit is valid when PSW.EBV = 0.	R	0

Note 1. The reset vector differs depending on the startup area. For details, see **Section 4, Address Space**.

(r) EBASE — Exception handler vector address register

This register indicates the exception handler vector address. This register is valid when the PSW.EBV bit is 1.

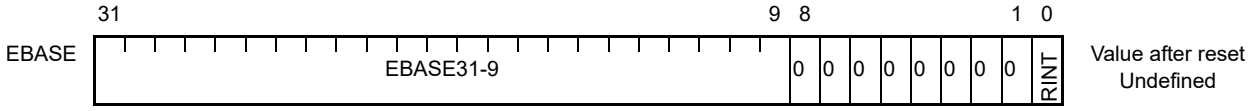


Table 3.103 EBASE Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 9	EBASE31-9	The exception handler routine address is changed to the address resulting from adding the offset address of each exception to the base address specified for this register. The EBASE8 to 0 bits are implicitly set to 0.	R/W	Undefined
8 to 1	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
0	RINT	When the RINT bit is set, the offset address of the exception handler used during interrupt servicing is decremented.	R/W	Undefined

CAUTION

For a CPU whose instruction addressing range is 128 MB, a value resulting from a sign-extension of bit 26 is automatically set to bits 31 to 27 of EBASE.

(s) INTBP — Base address of the interrupt handler table register

This register indicates the base address of the address table when the table reference method is selected as the interrupt handler address selection method.

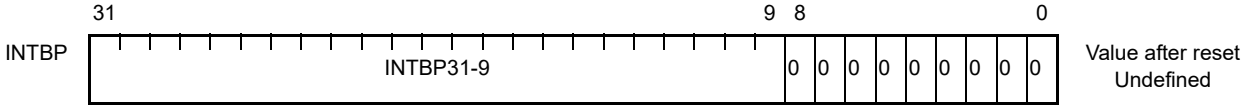


Table 3.104 INTBP Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 9	INTBP31-9	These bits indicate the base pointer address for an interrupt prescribed in the expanded specifications. The value indicated by these bits is the first address in the table used to determine the exception handler when the interrupt prescribed by the expanded specifications (EIINT0 to EIINT511) is acknowledged. The EBASE8 to 0 bits are implicitly set to 0.	R/W	Undefined
8 to 0	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0

CAUTION

For a CPU whose instruction addressing range is 128 MB, a value resulting from a sign-extension of bit 26 is automatically set to bits 31 to 27 of INTBP.

(t) PID — Processor ID register

The PID register retains a processor identifier that is unique to the CPU. The PID register is a read-only register.

CAUTION

The PID register indicates information used to identify the incorporated CPU core and CPU core configuration. Usage such that the software behavior varies dynamically according to the PID register information is not assumed.

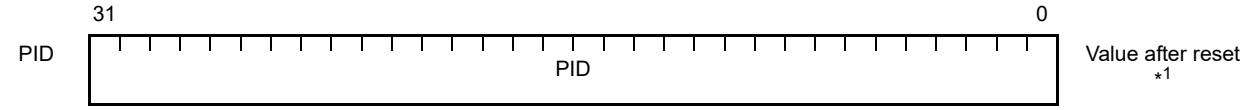


Table 3.105 PID Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 24	PID	Architecture identifier This identifier indicates the architecture of the processor.	R	$F5_H$
23 to 8		Function identifier This identifier indicates the functions of the processor. These bits indicate whether or not functions defined per bit are implemented (1: implemented, 0: not implemented). Bit 23 to 11 : Reserved Bit 10 : Double-precision floating-point operation function Bit 9 : Single-precision floating-point operation function Bit 8: Memory protection unit (MPU) function	R	0001_H
7 to 0		Version identifier This identifier indicates the version of the processor.	R	30_H

Note 1. The value is F500 0130_H in this product.

(u) SCCFG — SYSCALL operation setting register

This register is used to specify operations related to the SYSCALL instruction. Be sure to set an appropriate value to this register before using the SYSCALL instruction.

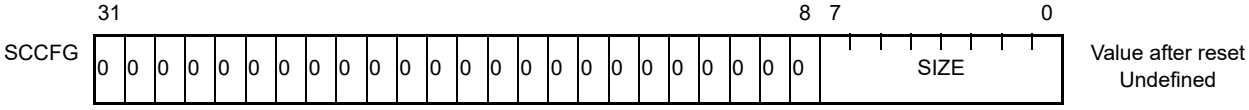


Table 3.106 SCCFG Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 8	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
7 to 0	SIZE	These bits specify the maximum number of entries of a table that the SYSCALL instruction references. The maximum number of entries the SYSCALL instruction references is 1 if SIZE is 0, and 256 if SIZE is 255. By setting the maximum number of entries appropriately in accordance with the number of functions branched by the SYSCALL instruction, the memory area can be effectively used. If a vector exceeding the maximum number of entries is specified for the SYSCALL instruction, the first entry is selected. Place an error processing routine at the first entry.	R/W	Undefined

(v) SCBP — SYSCALL base pointer register

The SCBP register is used to specify a table address of the SYSCALL instruction and generate a target address. Be sure to set an appropriate value to this register before using the SYSCALL instruction.

Be sure to set a word address to the SCBP register.

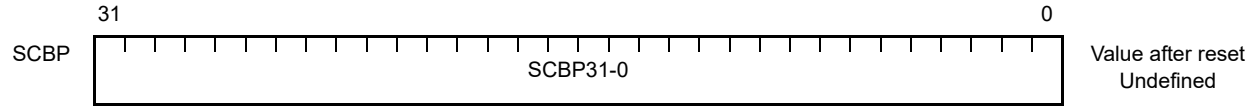


Table 3.107 SCBP Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 2	SCBP31-2	These bits indicate the base pointer address of the SYSCALL instruction. These bits indicate the first address in the table used by the SYSCALL instruction.	R/W	Undefined
1, 0	SCBP1-0	These bits indicate the base pointer address of the SYSCALL instruction. These bits indicate the first address in the table used by the SYSCALL instruction. Always set this bit to 0.	R	0

CAUTION

For a CPU whose instruction addressing range is 128 MB, a value resulting from a sign-extension of bit 26 is automatically set to bits 31 to 27 of SCBP.

(w) MCFG0 — Machine configuration register

This register indicates the CPU configuration.

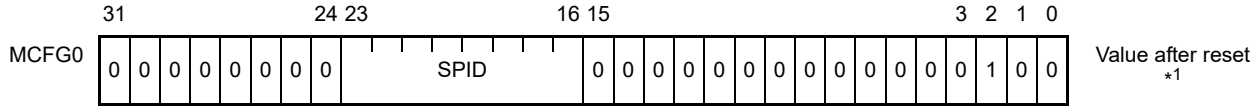


Table 3.108 MCFG0 Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 24	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
23 to 16	SPID	Bit 23 to 18: These are not supported in this product. (Reserved for the future expansion. Be sure to clear to 0) Bit 17, 16: These bits indicate the system protection number.	R/W	*2
15 to 3	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
2	—	(Reserved for future expansion. Be sure to set this bit to 1.)	R	1
1, 0	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0

Note 1. The value is 0003 0004_H in PCU (PE3) of this product.
Note 2. The value is 03_H in PCU (PE3) of this product.

(x) MCTL — Machine control register

This register is used to control the CPU.

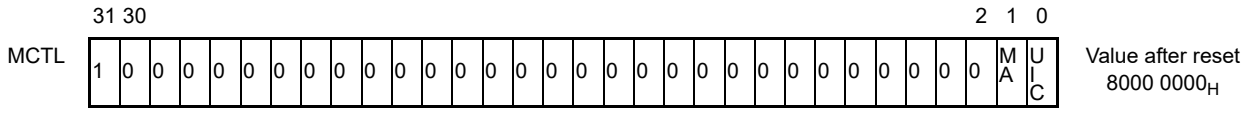


Table 3.109 MCTL Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31	EN	(Reserved for future expansion. Be sure to set this bit to 1.)	R	1
30 to 2	MT	(Reserved for future expansion. Be sure to clear to 0.)	R	0
1	MA	This bit specifies the operation when a misaligned access occurs. 0: An exception is generated in response to a misaligned access.*1 1: A misalignment exception is not generated and access still proceeds.*1	R/W	0
0	UIC	This bit is used to control the interrupt enable/disable operation in user mode. When this bit is set to 1, executing the EI/DI instruction in user mode becomes possible.	R/W	0

Note 1. This bit does not control the generation of misalignment exceptions in response to access by double-word instructions.

(3) Interrupt function registers

Table 3.110 Interrupt Function System Registers

Register No. (reg.ID, sel.ID)	Symbol	Function	Access Permission
SR10, 2	ISPR	Priority of interrupt being serviced	SV
SR11, 2	PMR	Interrupt priority masking	SV
SR12, 2	ICSR	Interrupt control status	SV
SR13, 2	INTCFG	Interrupt function setting	SV

(a) ISPR — Priority of interrupt being serviced register

This register holds the priority of the EIINTn interrupt being serviced. This priority value is then used to perform priority ceiling processing when multiple interrupts are generated.

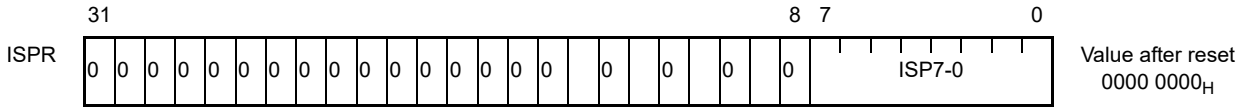


Table 3.111 ISPR Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 8	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
7 to 0	ISP7-0	These bits indicate the acknowledgment status of an EIINT*n* interrupt with a priority*1 that corresponds to the relevant bit position. 0: An interrupt request for an interrupt whose priority corresponds to the relevant bit position has not been acknowledged. 1: An interrupt request for an interrupt whose priority corresponds to the relevant position is being serviced by the CPU core. The bit positions correspond to the following priority levels: **Bit / Priority** 0 — Priority 0 (highest) 1 — Priority 1 : 6 — Priority 6 7 — Priority 7 When an interrupt request (EIINT*n*) is acknowledged, the bit corresponding to the acknowledged interrupt request is automatically set to 1. If PSW.EP is 0 when the EIRET instruction is executed, the bit with the highest priority among the ISP7 to 0 bits that are set (0 is the highest priority) is cleared to 0*2. While a bit in this register is set to 1, lower priority interrupts (EIINT*n*) are masked. Priority level judgment is therefore not performed when the system is determining whether to acknowledge an exception, meaning that exceptions will not be acknowledged. See *section 4.1.5 Interrupt Exception Priority and Priority Masking* in *RH850G3K User's Manual: Software*. When performing software-based priority control using the PMR register, be sure to clear this register by using the INTCFG.ISPC bit.	R*3	0

Note 1. See *section 4.1.5 Interrupt Exception Priority and Priority Masking* in *RH850G3K User's Manual: Software* for the details.

Note 2. Interrupt acknowledgment and auto-updating of values when the EIRET instruction is executed are disabled by setting the INTCFG.ISPC bit to 1. We recommend auto-updating, so in normal cases, clear the INTCFG.ISPC bit to 0.

Note 3. This is R or R/W, depending on the setting of the INTCFG.ISPC bit. It is recommended to use this register as a read-only (R) register.

(b) PMR — Interrupt priority masking register

This register is used to mask the specified interrupt priority.

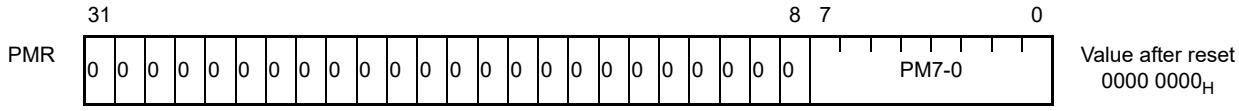


Table 3.112 PMR Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 8	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
7 to 0	PM7-0	These bits mask an interrupt request with a priority level that corresponds to the relevant bit position. 0: Servicing of an interrupt request with a priority that corresponds to the relevant bit position is enabled. 1: Servicing of an interrupt request with a priority that corresponds to the relevant bit position is disabled. The bit positions correspond to the following priority levels: **Bit** / **Priority** 0 / Priority 0 (highest) 1 / Priority 1 : / : 6 / Priority 6 7 / Priority 7 (lowest) While a bit in this register is set to 1, interrupts (EIINTn) with the priority corresponding to that bit are masked. Priority level judgment is therefore not performed when the system is determining whether to acknowledge an exception, meaning that exceptions will not be acknowledged*1.	R/W	0

Note 1. Specify the masks by setting the bits to 1 in order from the lowest-priority bit. For example, $F0_H$ can be set, but CC_H or $0F_H$ cannot.

(c) ICSR — Interrupt control status register

This register indicates the interrupt control status in the CPU.

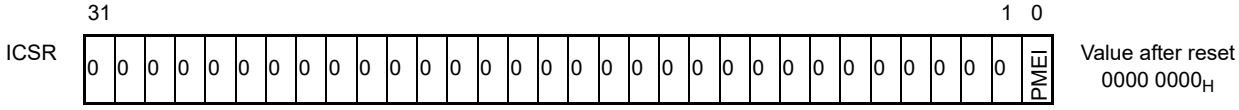


Table 3.113 ICSR Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 1	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
0	PMEI	This bit indicates that an interrupt (EIINTn) with the priority level masked by the PMR register exists	R	0

(d) INTCFG — Interrupt function setting register

This register is used to specify settings related to the CPU's internal interrupt function.

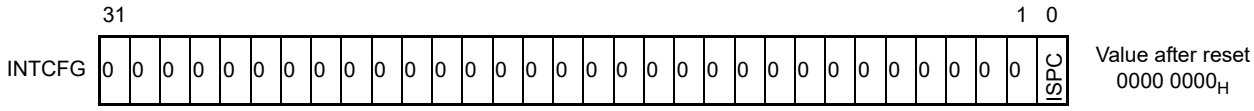


Table 3.114 INTCFG Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 1	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
0	ISPC	This bit changes how the ISPR register is written. 0: The ISPR register is automatically updated. Updates triggered by the program (via execution of LDSR instruction) are ignored. 1: The ISPR register is not automatically updated. Updates triggered by the program (via execution of LDSR instruction) are performed. If this bit is cleared to 0, the bits of the ISPR register are automatically set to 1 when an interrupt (EIINT*n*) is acknowledged, and cleared to 0 when the EIRET instruction is executed. In this case, the bits are not updated by an LDSR instruction executed by the program. If this bit is set to 1, the bits of the ISPR register are not updated by the acknowledgement of an interrupt (EIINT*n*) or by execution of the EIRET instruction. In this case, the bits can be updated by an LDSR instruction executed by the program. In normal cases, the ISPC bit should be cleared. When performing software-based priority control, however, set this bit to 1 and perform priority control by using the PMR register.	R/W	0

(4) FPU Function Registers

This product does not include an FPU.

(5) MPU Function Registers

(a) MPU function system registers

The MPU function system registers are read from or written to by using the LDSR or STSR instructions and specifying the system register number, which is made up of a register number and selection ID.

Table 3.115 MPU Function System Registers (1/2)

Register No. (reg.ID, sel.ID)	Symbol	Function	Access Permission
SR0, 5	MPM	Memory protection operation mode setting	SV
SR1, 5	MPRC	MPU region control	SV
SR4, 5	MPBRGN	MPU base region number	SV
SR5, 5	MPTRGN	MPU end region number	SV
SR11, 5	MCR	Memory protection setting check result	SV
SR0, 6	MPLA0	Protection area minimum address	SV
SR1, 6	MPUA0	Protection area maximum address	SV
SR2, 6	MPAT0	Protection area attribute	SV
SR4, 6	MPLA1	Protection area minimum address	SV
SR5, 6	MPUA1	Protection area maximum address	SV
SR6, 6	MPAT1	Protection area attribute	SV
SR8, 6	MPLA2	Lower address of the protection area	SV
SR9, 6	MPUA2	Protection area maximum address	SV
SR10, 6	MPAT2	Protection area attribute	SV
SR12, 6	MPLA3	Protection area minimum address	SV
SR13, 6	MPUA3	Protection area maximum address	SV
SR14, 6	MPAT3	Protection area attribute	SV

Table 3.115 MPU Function System Registers (2/2)

Register No. (reg.ID, sel.ID)	Symbol	Function	Access Permission
SR16, 6	MPLA4	Not implemented. A value of 0 is returned when read and writing is ignored.	SV
SR17, 6	MPUA4		SV
SR18, 6	MPAT4		SV
SR20, 6	MPLA5		SV
SR21, 6	MPUA5		SV
SR22, 6	MPAT5		SV
SR24, 6	MPLA6		SV
SR25, 6	MPUA6		SV
SR26, 6	MPAT6		SV
SR28, 6	MPLA7		SV
SR29, 6	MPUA7		SV
SR30, 6	MPAT7		SV
SR0, 7	MPLA8		SV
SR1, 7	MPUA8		SV
SR2, 7	MPAT8		SV
SR4, 7	MPLA9		SV
SR5, 7	MPUA9		SV
SR6, 7	MPAT9		SV
SR8, 7	MPLA10		SV
SR9, 7	MPUA10		SV
SR10, 7	MPAT10		SV
SR12, 7	MPLA11		SV
SR13, 7	MPUA11		SV
SR14, 7	MPAT11		SV
SR16, 7	MPLA12		SV
SR17, 7	MPUA12		SV
SR18, 7	MPAT12		SV
SR20, 7	MPLA13		SV
SR21, 7	MPUA13		SV
SR22, 7	MPAT13		SV
SR24, 7	MPLA14		SV
SR25, 7	MPUA14		SV
SR26, 7	MPAT14		SV
SR28, 7	MPLA15		SV
SR29, 7	MPUA15		SV
SR30, 7	MPAT15		SV

(1) MPM — Memory protection operation mode register

The memory protection mode register is used to define the basic operating state of the memory protection function.

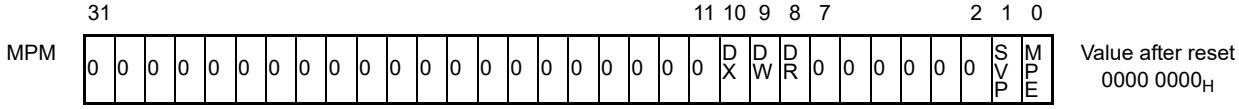


Table 3.116 MPM Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 11	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
10	DX	This bit specifies the default operation when an instruction is executed at an address that does not exist in a protection area. 0: Disable executing an instruction at an address that does not exist in a protection area. 1: Enable executing an instruction at an address that does not exist in a protection area.	R/W	0
9	DW	This bit specifies the default operation when writing to an address that does not exist in a protection area. 0: Disable writing to an address that does not exist in a protection area. 1: Enable writing to an address that does not exist in a protection area.	R/W	0
8	DR	This bit specifies the default operation when reading from an address that does not exist in a protection area. 0: Disable reading from an address that does not exist in a protection area. 1: Enable reading from an address that does not exist in a protection area.	R/W	0
7 to 2	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
1	SVP*3, *4	In SV mode (when PSW.UM = 0), this bit is used to specify whether to restrict access according to the SX, SW, and SR bits of the MPAT register for each protection area.*1 0: As usual, implicitly enable all access in SV mode. 1: Restrict access according to the SX, SW, and SR bits even in SV mode.*2	R/W*4	0
0	MPE	This bit is used to specify whether to enable or disable MPU function. 0: Disable 1: Enable	R/W	0

Note 1. When the SVP bit is set to 1, access will become restricted after a certain number of clock cycles, even in SV mode. Therefore, specify the protection area before setting the SVP bit to prevent restriction of access by the program itself.

Note 2. If access is restricted in SV mode., execution of MDP exceptions or the MIP exception handling itself might not be possible depending on the settings. Be careful to specify settings so that access to the memory area necessary for the exception handler and exception handling is permitted.

Note 3. Access by the CPU in supervisor mode is not restricted during the processing of SYSCALL instructions while SVP = 1 and in table references in response to EIINT interrupts involving table reference.

Note 4. When the SVP bit is set to 1 to use the debugger, enable execution of instruction by SV mode in the FE60 0000_H to FE7F $FFFF_H$ area. When a debugger manufactured by a third party executes instructions by downloading code to the global RAM area or local RAM area of PE1, enable instructions for SV mode in said areas.

(2) MPRC — MPU region control register

Bits used to perform special memory protection function operations are located in this register.

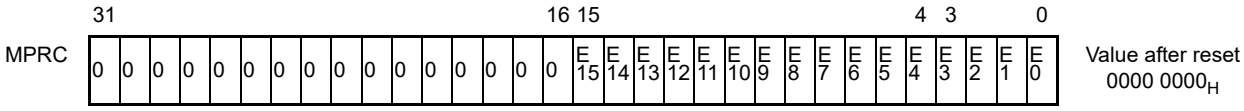


Table 3.117 MPRC Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 16	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
15 to 4	E15-E4	These bits are not supported in this product. (Reserved for future expansion. Be sure to clear to 0.)	R	0
3 to 0	E15-E0	These are the enable bits for each protection area. Bit En is a copy of bit MPATn.E (where n = 3 to 0). The number of the protection areas in this product is 4.	R/W	0

(3) MPBRGN — MPU base region register

This register indicates the minimum usable MPU area number.

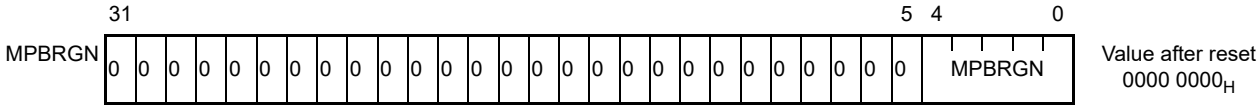


Table 3.118 MPBRGN Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 5	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
4 to 0	MPBRGN	These bits indicate the smallest number of an MPU area.	R	0000

(4) MPTRGN — MPU end region register

This register indicates the maximum usable MPU area number.

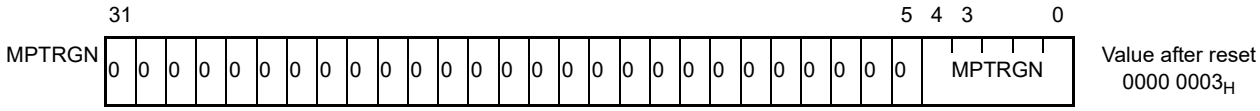


Table 3.119 MPTRGN Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 5	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
4 to 0	MPTRGN	These bits indicate the largest number of an MPU area.	R	0011

(5) MCR — Memory protection setting check result

This product does not incorporate the memory protection setting function.

This register is reserved for the future expansion.

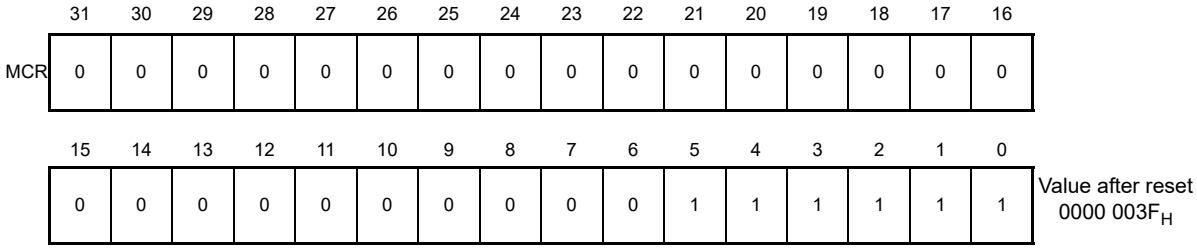


Table 3.120 MCR Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 0	—	(Reserved for future expansion)	R	0000 003F$_H$

(6) MPLAn — Protection area minimum address register

These registers indicate the minimum address of area n (n = 0 to 3).

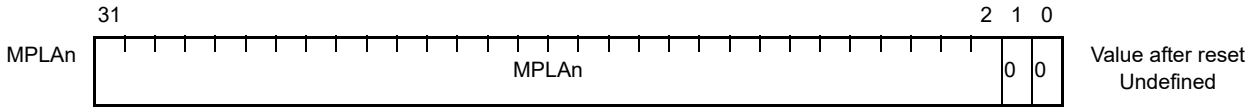


Table 3.121 MPLAn Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 2	MPLA31-2	These bits indicate the minimum address of area n. The MPLA1-0 bits are implicitly set to 0.	R/W	Undefined
1, 0	—	Reserved for future expansion. Be sure to clear these bits to 0.	R	0

CAUTION

For a CPU whose instruction addressing range is 128 MB, a value resulting from a sign-extension of bit 26 is automatically set to bits 31 to 27 of MPLAn.

(7) MPUAn — Protection area maximum address register

These registers indicate the maximum address of area n (n = 0 to 3).

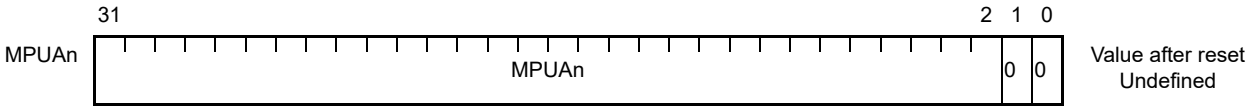


Table 3.122 MPUAn Register Contents

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 2	MPUA31-2	These bits indicate the maximum address of area n. The MPUA1-0 bits are implicitly set to 1.	R/W	Undefined
1, 0	—	Reserved for future expansion. Be sure to clear these bits to 0.	R	0

CAUTION

For a CPU whose instruction addressing range is 128 MB, a value resulting from a sign-extension of bit 26 is automatically set to bits 31 to 27 of MPUAn.

(8) MPATn — Protection area attribute register

These registers indicate the attributes of area n (where n = 0 to 3).

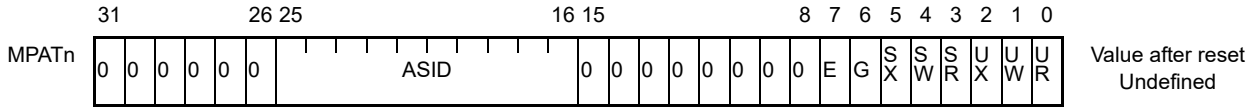


Table 3.123 MPATn Register Contents (1/2)

Bit Position	Bit Name	Function	R/W	Value after Reset
31 to 26	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
25 to 16	ASID	These bits indicate the ASID value to be used as the area match condition.	R/W	Undefined
15 to 8	—	(Reserved for future expansion. Be sure to clear to 0.)	R	0
7	E	This bit indicates whether area n is enabled or disabled. 0: Area n is disabled. 1: Area n is enabled.	R/W	0
6	G	0: ASID match is the condition. 1: ASID match is not the condition. When this bit is 0, the condition of the area match is MPATn.ASID = ASID.ASID. When this bit is 1, the area match of the values of MPATn.ASID and ASID.ASID is the condition.	R/W	Undefined
5	SX	This bit indicates the execution privilege for supervisor mode.*1 0: The execution is disabled. 1: The execution is enabled.	R/W	Undefined
4	SW	This bit indicates the write permission for supervisor mode.*1 0: Writing is disabled. 1: Writing is enabled.	R/W	Undefined
3	SR	This bit indicates the read permission for supervisor mode*1 0: Reading is disabled. 1: Reading is enabled.	R/W	Undefined

Table 3.123 MPATn Register Contents (2/2)

Bit Position	Bit Name	Function	R/W	Value after Reset
2	UX	This bit indicates the execution privilege for user mode. 0: The execution is disabled. 1: The execution is enabled.	R/W	Undefined
1	UW	This bit indicates the write permission for user mode. 0: Writing is disabled. 1: Writing is enabled.	R/W	Undefined
0	UR	This bit indicates the read permission for user mode. 0: Reading is disabled. 1: Reading is enabled.	R/W	Undefined

Note 1. If access is restricted in SV mode, execution of MDP exceptions or the MIP exception handling itself might not be possible depending on the settings. Be careful to specify settings so that access to the memory area necessary for the exception handler and exception handling is permitted.

(6) Cache operation function registers

The RH850/E1x does not have cache processing function registers.

(7) Data buffer operation function registers

The RH850/E1x does not have data buffer operation function registers.

3.3.2 Inter-Processor Interrupts

The PCU includes an inter-processor interrupt control register (IPIR_CH0-3). It is a CPU peripheral and its functions and addresses are the same as those of the CPU1. IPIR_CH0 to IPIR_CH3 registers are assigned to each PE which are accessible only from their own PE. For details, see **Section 3.2.3, Inter-Processor Interrupts**.

3.3.3 Reliability Functions

3.3.3.1 System Error Notification Control Function (SEG)

Errors due to an instruction fetch or data access can be the sources of system error exceptions. A SYSERR exception is an FE level exception from which return or recovery is not possible.

For source codes (FEIC) of the SYSTEM exceptions and error handling, see **Table 3.125, Error Factor Codes and Handling of G3K Core SYSERR Exceptions**. When a system error exception is accepted, the PSW.EBV bit is cleared to 0 and the exception handler always refers to RBASE.

SEG (SysErrGen) controls interrupt notification and recording after a system error occurred by a data access.

(1) Register Set

(a) SEG_CONT — System Error Control Register

This register enables and disables the error notification of each SYSERR source. When the bit is set to 1, SYSERR notification of error occurrence is enabled. When it is cleared to 0, SYSERR notification is disabled although the error flag in response to the error factor occurrence is set.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—									RMWE	—	RAME	—	EXTE	FCHE	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R	R/W	R	R/W	R/W	R

Table 3.124 SEG_CONT Register Contents

Bit Position	Bit Name	Function
15 to 7	—	Reserved These bits are always read as 0. The write value should always be 0.
6	RMWE	Local RAM Read/Modify/Write Error Notification Enable: This bit enables SYSERR notification of an ECC error that occurs when a read access is made to the local RAM during read/modify/write operation after a byte or half word write is made to the local RAM.
5	—	Reserved This bit is always read as 0. The write value should always be 0.
4	RAME	Local RAM ECC Error Notification Enable: This bit enables the SYSERR notification of an ECC error occurred when the CPU accesses the data in the local RAM area.
3	—	Reserved This bit is always read as 0. The write value should always be 0.
2	EXTE	P-Bus/H-Bus/GRAM Area Error Notification Enable: This bit enables SYSERR notification of an error occurring when the P-Bus/H-Bus/GRAM area is accessed. Such errors include: • P-Bus: Data Parity Error, Guard Error • H-Bus: Error response in the slave modules • GRAM: Data ECC Error, Address Parity Error, Guard Error
1	FCHE	Code Flash Error Notification Enable: This bit enables the SYSERR notification of an error occurred when the CPU accesses the data in the code flash. Such errors include: • Data ECC Error • Address Parity Error
0	—	Reserved This bit is always read as 0. The write value should always be 0.

Table 3.125 Error Factor Codes and Handling of G3K Core SYSERR Exceptions

Factor Code	Error Contents
10	Reserved
11	Instruction fetch errors
12	Reserved
13	Reserved
14	Errors in which their notification is enabled by the SEG_CONT register. Errors that have occurred can be identified by reading the SEG_FLAG register
15	Reserved
16	Reserved
17	Reserved
18	Reserved
19	Reserved
1A	Reserved
1B	Reserved
1C	Reserved
1D	Reserved
1E	Reserved
1F	Reserved

(b) SEG_FLAG — System Error Flag Register

This register indicates the flags to store the status under which each SYSERR factor occurs. When a SYSERR factor occurs, the corresponding error factor flag is set to 1. The SYSERR can also be generated by direct writing to the register and setting the flag to 1. Each flag can be cleared by writing 0 after reading the state of 1.

Even if 0 is written in a subsequent write operation to a flag already set to 0 during reading, the flag is set in response to an error occurring between the read and the write.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	RMWF	—	RAMF	—	EXTF	FCHF	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R	R/W	R	R/W	R/W	R

Table 3.126 SEG_FLAG Register Contents

Bit Position	Bit Name	Function
15 to 7	—	Reserved These bits are always read as 0. The write value should always be 0.
6	RMWF	Flag corresponding to bit 6 of the SEG_CONT register
5	—	Reserved This bit is always read as 0. The write value should always be 0.
4	RAMF	Flag corresponding to bit 4 of the SEG_CONT register
3	—	Reserved This bit is always read as 0. The write value should always be 0.
2	EXTF	Flag corresponding to bit 2 of the SEG_CONT register
1	FCHF	Flag corresponding to bit 1 of the SEG_CONT register
0	—	Reserved This bit is always read as 0. The write value should always be 0.

3.4 Inter-CPU Functions

3.4.1 Processor Element Identifier

The PEID, each processor element ID number, can be read from the PEID field in the HTCFG0 register. Which CPU core performs a specific program can be understood by referring to the PEID. The following shows the PEID of this product.

CPU core	PEID
CPU1 (PE1)	001_B
PCU (PE3)	011_B

3.4.2 Inter-Processor Interrupt Function

Both CPU1 and the PCU individually have IPIR registers as their own peripheral functions. Setting of the IPIR register enables an EI-level interrupt request from a PE to another PE. For details, see **Section 3.2.3, Inter-Processor Interrupts**.

3.4.3 Exclusive Function

The local RAM, global RAM, and exclusive control register (MEV) are available as a resource for exclusive control. As atomic operation instruction, the instructions of LDL/STC, CAXI, SET1, CLR1, and NOT1 can be performed for the local RAM and global RAM, and the instructions of CAXI, SET1, CLR1, and NOT1 can be performed for the exclusive control register (MEV). These registers are also accessible by the LD and ST instructions but not regarded as atomic operation.

CAUTION

Access to the local RAM of the PCU or an MEV register by using the CAXI, SET1, CLR1, or NOT1 instructions are atomic operations, but access to the local RAM of CPU1 or global RAM by the same instructions is not. Use the MEV for exclusive control of PCU and CPU1.

3.4.3.1 Exclusive Control Register (MEV)

This register supports exclusive control for variables shared between PEs (common resources).

(MEV = Mutual Exclusion Variable Register)

- 32 32-bit MEV registers are included.
- 1-, 8-, 16-, and 32-bit accesses are available for each MEV.
- Accesses from CPU1 (PE1) and the PCU (PE3) can be made.
- Atomic operation instructions of CAXI, SET1, CLR1, and NOT1 can be performed.

CPU1 and the PCU each have an independent access path for the MEV registers. Therefore, when CPU1 and the PCU each access different MEV registers, they do not need to wait for access. When they access the same MEV register, however, waiting for access is required.

Table 3.127 Base Address of SEG Register: FFFE EC00$_H$

Register Symbol	Register Name	R/W	Value after Reset	Operable Bit				Access Size
				1	8	16	32	
G0MEV0	Exclusive Control Register 0	R/W	0000 0000$_H$	√	√	√	√	+00$_H$
G0MEV1	Exclusive Control Register 1	R/W	0000 0000$_H$	√	√	√	√	+04$_H$
G0MEV2	Exclusive Control Register 2	R/W	0000 0000$_H$	√	√	√	√	+08$_H$
G0MEV3	Exclusive Control Register 3	R/W	0000 0000$_H$	√	√	√	√	+0C$_H$
:	:	:	:	:	:	:	:	:
G0MEV31	Exclusive Control Register 31	R/W	0000 0000$_H$	√	√	√	√	+7C$_H$

3.4.3.2 Operation of the LDL.W and STC.W Instructions

The LDL.W and STC.W instructions can be used to obtain atomic read-modify-write operations for accurate processing in the updating of memory by multicore systems. The LDL.W and STC.W instructions operate as follows. The LDL.W and STC.W instructions are only in the instruction set for the CPU (G3M) and not in that for the PCU (G3K). For the operation of the LDL.W and STC.W instructions, refer to the *RH850G3M User's Manual: Software*.

- Link generation: The CPU is capable of generating links to both the local RAM and global RAM. Executing the LDL.W instruction on the target RAM for the operation leads to link address being registered, the link flag being set, and a link being generated in response to reading by the instruction. Two link flag systems are provided, one each for the following two areas of RAM.

  (1) The local RAM for the given processor: 1

  (2) The global RAM: 1

  Since these link flags are generated independently of each other, a link that was previously generated (e.g. by the CPU to the local RAM) is not lost when the CPU executes the LDL.W instruction for the other RAM (e.g. the global RAM).

- Success in storing: After a link has been generated, storing will only proceed in response to executing an STC.W instruction corresponding to the generated link.
- Failure in storing: If a link is lost, storing does not proceed even when an STC.W instruction for the corresponding address is processed. Storing also does not proceed when an STC.W instruction that does not correspond to the link is processed.
- Condition for successful storing: If the following condition is met, the STC.W instruction is judged to be for the address corresponding to the link.
  - The address for the LDL.W instruction which generated the link matches that for the STC.W instruction.
- Loss of the link: If any of the following occurs, the link flag is cleared and the link is lost.
  - Any of the following processing by the CPU for which the link was generated:
    - An STC.W instruction being executed. The corresponding link (for (1) or (2) above) will be lost irrespective of the success or failure of storing.
    - Occurrence of an exception or the CPU executing an instruction to return from an exception processing routine (FERET or EIRET). The link for the linked RAM area and the link flag are cleared.
    - A successive LDL.W instruction being executed for a location corresponding to the link flag for the linked RAM.
      The link generated in response to the preceding LDL.W instruction will be lost. Do not execute such processing.
    - Storing operations other than execution of a STC.W instruction for the address* indicated by the link. Do not execute such processing.

– Access as described below by another bus master:
  – Any storing operation, including execution of an STC.W instruction for the address* indicated by the link. The corresponding link will be lost.

Note. Refers to the addresses of locations that match the 27 higher-order address bits of the address for which the link was created.

Success of the STC.W instruction means that the LDL.W and STC.W instructions have realized an atomic Read-Modify-Write operation.

3.5 Usage Notes

3.5.1 Synchronization of Store Instruction Completion and Subsequent Instruction Generation

When a control register is updated by a store instruction, there is a time lag from execution of the store instruction by the CPU to actual updating of the control register. Therefore, appropriate synchronization processing is required to ensure the control register reflects updated contents before execution of a subsequent instruction. How to perform synchronization processing is shown below.

For details of the procedure regarding updating of the system registers by the LDSR instruction and synchronization with subsequent instructions, refer to APPENDIX A, HAZARD RESOLUTION PROCEDURE FOR SYSTEM REGISTERS in the *RH850G3M User's Manual: Software or the RH850G3K User's Manual: Software*.

3.5.1.1 When updated results in the control registers are reflected in the implementation of a subsequent instruction:

Example 1: An interrupt may be enabled by implementation of an EI instruction after an interrupt request is cleared by access from the control register in the INTC2 and the peripheral circuits. Proceed as follows in this case.

(1) Store instruction to update a control register (ST.W, etc.)

(2) Dummy reading of the above-mentioned control register (LD.W, etc.)*1

(3) SYNCP

(4) Subsequent instruction (EI, etc.)

Example 2: Implement the same processing even when the access required after waiting to secure the updating of a given control register (register A) is to another control register (register B). This includes the following cases: the interlinked operation of different peripheral modules and when releasing the interrupt mask in INTC after making peripheral module settings.
However, this processing is unnecessary if control registers A and B are in the same peripheral group.

For the correspondence between the peripheral groups and the peripheral modules/registers, see **Appendix, List of Registers**.

(1) Issue the instruction for storage to update control register A (TS.W, etc.)

(2) Dummy-read the above control register (LD.W, etc.)*1

(3) Issue SYNCP.

(4) Issue the instruction to access control register B (ST.W, LD.W, etc.)

The same processing is also required when access to control registers and memory within the scope of protection starts after waiting for the completion of settings for safety functions such as memory protection, ECC checking, and so on.

Note 1. Dummy reading of any register of the same peripheral group can be used instead.

3.5.1.2 When the updated results in the control registers and memories are reflected in the instruction fetch of a subsequent instruction:

(a) If you wish to write an instruction to the RAM and then branch to the RAM to execute the written instruction, handle this as follows.

(1) Store instruction to update a memory (ST.W, etc.)

(2) Dummy read of the above-mentioned memory (LD.W, etc.)

(3) SYNCP

(4) SYNCI

(5) Subsequent instruction (branch instruction, etc.)

(b) When branching to a target memory after waiting for the completion of updating the control registers for memory protection and ECC, handle this as follows.

(1) Store instruction to update a control register (ST.W, etc.)

(2) Dummy reading of the control register (LD.W, etc.)

(3) SYNCP

(4) SYNCI

(5) Subsequent instruction (branch instruction, etc.)

3.5.1.3 When switching the code flash area

In this case, refer to (7) Updating the FCUFAREA Register, in section 9, Usage Notes, in the *RH850/E1x Flash Memory User's Manual: Hardware Interface*.

3.5.1.4 When executing the SYNCM instruction to wait for the completion of update by the store instruction

When executing the SYNCM instruction to wait for the completion of updating the destination for access by the preceding store instruction, the CPU that executes the store instruction and the destination for access must be as follows.

CPU that Executes the Store Instruction	Destination for Access by the Store Instruction
CPU1	Local RAM (self), Local RAM (PE1), Global RAM

3.5.2 Accesses to Registers by Bit-Manipulation Instructions

Processing of a bit-manipulation instruction takes the form of atomic reading, modification, and writing of an eight-bit unit. Thus, access by a bit-manipulation instruction is only possible for registers for which reading and writing in 8-bit units is possible. However, take care in the cases of registers that contain multiple flag bits, since the read-modify-write cycle may also clear flags other than that which was the target for clearing.

Write access to the FlexRay registers by using bit-manipulation instructions is not atomic. Access by other masters may interrupt the read-modify-write processing of these instructions.

3.5.3 Ensuring Coherency after Code Flash Programming

The CPU has an efficient instruction cache and data buffer for the code flash area.

Therefore, after using self-programming to program the code flash memory, clear the instruction cache and data buffer to ensure coherency. The instruction cache and data buffer can be cleared by using the ICCTRL register and the CDBCR register, respectively. The instruction cache and data buffer can also be cleared at the same time by using the TM_CC register. For details of the TM_CC register, see **Section 35.4.5, TM_CC — Cache Clear Operation Register**.

When the PCU is in use, the TM_CC register can also be used in the same way to clear the cache and buffer after programming the code flash memory.

3.5.4 Overwriting Context when Acknowledging Multiple Exceptions

Exceptions may be acknowledged regardless of the states of the ID or NP bits of the PSW register. This depends on the type of exception. When multiple exceptions occur, the contents of the system registers which hold the context information are overwritten. Regarding the conditions for acknowledging exceptions from each source and the possibility of return and recovery, see the list of exception sources in *RH850G3M User's Manual: Software* and *RH850G3K User's Manual: Software*.

3.5.5 Usage Notes on Prefetching

CPU executes speculative instruction fetching from locations later than the current value of the program counter to maintain the throughput of instruction fetches. Reading from memory due to such prefetching may proceed even from locations to which instruction codes have not been assigned (note 1 in **Figure 3.5**). Please note the following. The CPU does not execute values read in such cases.

These notes apply to instruction fetching from memory in general.

- Occurrence of ECC errors due to values in memory being undefined

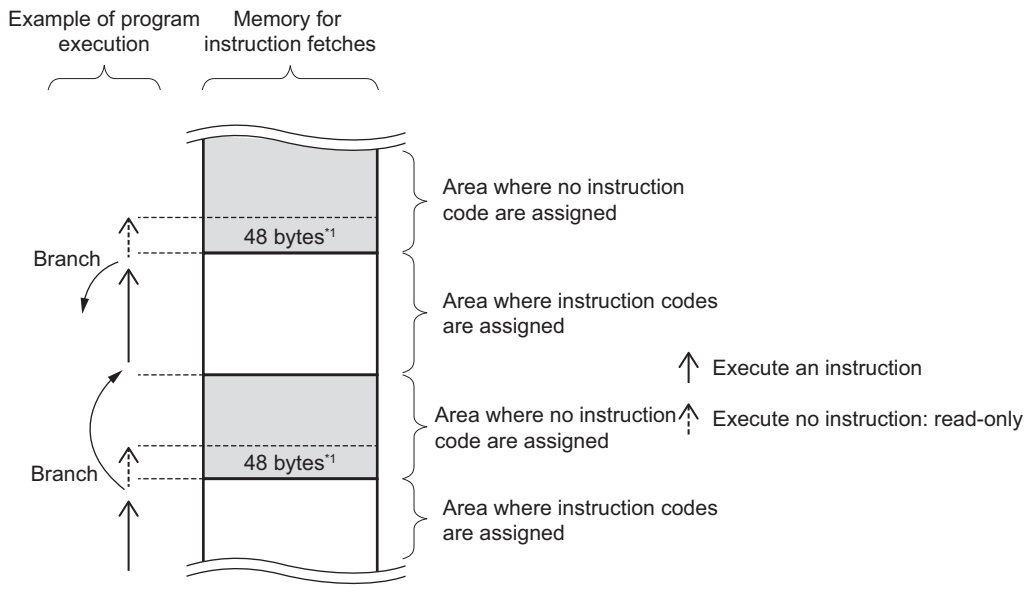
  This prefetching may lead to an ECC error in case of reading from the code flash memory after it has been erased or from the local RAM or global RAM before initialization. When instruction codes are assigned to memory, initialize said area with values as desired (note 1 in **Figure 3.5**).

- Detection of illegal access by the GRG or IPG

  The GRG or IPG may detect such prefetching as illegal access. To prevent prefetching being detected as an illegal access, do not allow any region of overlap area with said areas (note 1 in **Figure 3.5**) and areas to which access is prohibited by the GRG or IPG. Reading from an area protected by the MPU does not cause a memory protection exception.

- Access to Access Prohibited Area

  Assign instruction codes to memory without allowing any overlap between said area (note 1 in **Figure 3.5**) and an access-prohibited area.



Note 1. Area from final address of instruction code to final address + 48 bytes. Even if instruction codes are not placed in this area, requests for prefetching may lead to reading from memory at the time of branching.

Figure 3.5 Area that Requires Attention Regarding Prefetching

Section 4 Address Space

Table 4.1 shows the address space of the RH850/E1M-S.

When making an access to the on-chip I/O register space, access the addresses shown in Appendix, List of Registers. Do not access an address that is not specified in **Table 4.1** and an area where any access is prohibited. If an unspecified address or reserved area is accessed, operation is not guaranteed.

4.1 Address Space

Table 4.1 Address Space

Address	Address Space Type	Size
0000 0000_H to 003F $FFFF_H$ (0001 7000_H to 0001 $7FFF_H$)	Code flash (user area read) (FCU firmware area (Map is switched by FCUFAREA register))*3	4 MB (4 KB)
0040 0000_H to 00FF $FFFF_H$	Access prohibited	
0100 0000_H to 0100 $7FFF_H$	Code flash (user boot area read)	32 KB
0100 8000_H to 0FFF $FFFF_H$	Access prohibited	
1000 0000_H to 1FFF $FFFF_H$	On-chip I/O register	256 MB
2000 0000_H to FEBE $FFFF_H$	Access prohibited	
FEBF 0000_H to FEBF $FFFF_H$	Local RAM (PE1)	64 KB
FEC0 0000_H to FEDE $FFFF_H$	Access prohibited	
FEDF 0000_H to FEDF $FFFF_H$	Local RAM (self) *1	64 KB
FEE0 0000_H to FEED $FFFF_H$	Access prohibited	
FEEE 0000_H to FEF1 $FFFF_H$	Global RAM area	256 KB
FEF2 0000_H to FEFF $FFFF_H$	Access prohibited	
FF00 0000_H to FFFD $FFFF_H$ (FF20 0000_H to FF20 $FFFF_H$) (FFA1 2000_H to FFA1 $2FFF_H$)	On-chip I/O register (Data flash (read/write)) (FCU RAM)	16 MB to 128 KB (64 KB) (4 KB)
FFFE 0000_H to FFFE $DFFF_H$	Access prohibited	
FFFE $E000_H$ to FFFE $FFFF_H$	On-chip I/O register (self*2)	8 KB
FFFF 0000_H to FFFF $4FFF_H$	Access prohibited	
FFFF 5000_H to FFFF $FFFF_H$	On-chip I/O register	44 KB

Note 1. When accessing a local RAM from PE1 or other unit, accessing the local RAM (PE1) allows access to the local RAM in PE1 and accessing the local RAM (self) allows access to the local RAM in each PE (PE1/PCU). The local RAM (self) areas have the following differences because the local RAM (self) capacity differs between PE1 and the PCU.
- PE1 local RAM (self): FEDF 0000_H to FEDF $FFFF_H$ (64 KB)
- PCU local RAM (self): FEDF 8000_H to FEDF $FFFF_H$ (32 KB)

Also note that some tools do not support access to the local RAM of PE1 via the local RAM (self) area. If this is the case, access to the local RAM area for PE1 is via the local RAM (PE1) area and access to the local RAM of the PCU is via the local RAM (self) area.

Note 2. The on-chip I/O register (self) has CPU peripheral function. CPU peripheral function is typically unique to each PE and no access is allowed to the CPU peripheral of other PE. However, the exclusive control register (MEV) is an exception and shares the function between PEs.

Note 3. For details, see the *RH850/E1x Flash Memory User's Manual: Hardware Interface*.

4.2 Address Space Viewed from Each Bus Master

Figure 4.1 shows address spaces viewed from each bus master.

4.2.1 Space in which instructions can be fetched

1. Instructions of PE1 and the PCU can be fetched from the code flash, local RAM and global RAM.
2. The reset vector (RBASE initial value) of PE1 and the PCU:

- When starting up from the user boot area, its head address is 0100 0000_H.
- When starting up from the user area, its head address is 0000 0000_H.

4.2.2 Data space accessible by PE1

All spaces are accessible.

4.2.3 Data space accessible by PCU

All spaces are accessible. However, an access from PE1 to the on-chip I/O register, 1000 0000_H to 1FFF $FFFF_H$ is not available. Use FC00 0000_H to FE5F $FFFF_H$ for such access.

4.2.4 Data space accessible by DMA (DMAC, DTS)

See **Figure 4.1** for the accessible spaces from the DMA.

4.2.5 Data space accessible by H-Bus

See **Figure 4.1** for the accessible spaces from the H-Bus.

Figure 4.1 Address Space Viewed from Each Bus Master

4.3 Global RAM and Retention RAM

The global RAM is divided into two, bank A and bank B. Different banks can be accessed concurrently. The upper 32 Kbytes of bank A is the retention RAM that retains data even during power-off standby state.

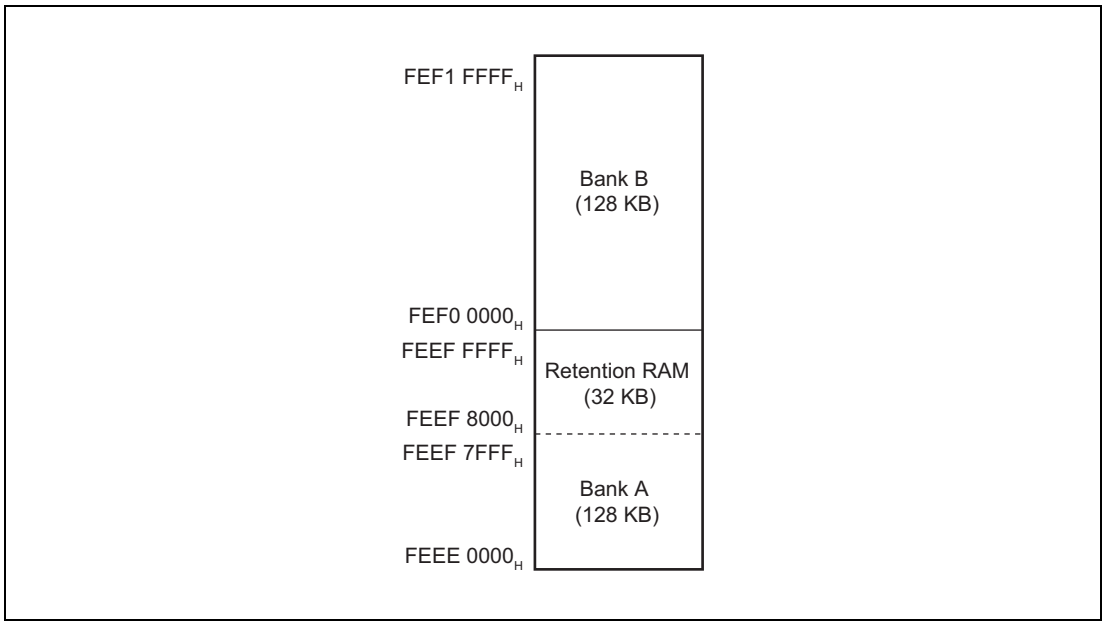


Figure 4.2 Global RAM

Section 5 Operating Mode

5.1 Features

- This LSI has three mode pins to determine the operating mode (MD0, MD1, and FLMODE).

5.2 Operating Mode

This LSI has multiple operating modes, which can be selected with the three pins (MD1, MD0, and FLMODE) and the setting of STMSEL1/STMSEL0 in option byte 0. For the procedures to set STMSEL1/STMSEL0, see **Section 33, Flash Memory**. **Table 5.1** shows the list of the operating modes.

Table 5.1 Selection of Operating Mode

Value Set in the Pin			Value Set in Option Byte 0 OPBT0		Operating Mode	Startup Area	Types of I/F*1	Remarks
MD1	MD0	FLMODE	STM SEL1	STM SEL0				
0	0	0	0	0	User boot mode	User area	It is possible to select I/F by OPBT2 in option byte. See **Section 33.10.2, OPBT2 — Option Byte 12 to 9 Bits Arrangement** for the details.	On-chip debug is available.
			0	1	User boot mode	User boot area		
			1	×	Serial programming mode	Boot area	Writer I/F (2-line UART)	Serial programming is available
0	0	1	×	×	Boundary scan mode	—	JTAG	Boundary scan is available.
0	1	0	×	×	Serial programming mode	Boot area	Writer I/F (2-line UART)	Serial programming is available
0	1	1	×	×	Serial programming mode	Boot area	Writer I/F (3-line clock synchronization)	Serial programming is available

Note: X = don't care.

Note 1. For the correspondence between the pin function and pin state in each interface, see Section 2.5.3, Pin State.

5.2.1 User Boot Mode

After release from the reset state, instruction fetch is carried out from the user boot area or the user area.

5.2.2 Serial Programming Mode

After release from the reset state, the LSI boots up from the on-chip boot program and starts connection in the specified transmission method. For details, see **Section 33, Flash Memory**.

5.2.3 Boundary Scan Mode

This is a mode to use the boundary scan function in accordance with the standard of IEEE1149.1. For details, see **Section 36, Boundary Scan**.

Section 6 Interrupt

The interrupt controller (INTC) determines priority of interrupt sources and controls interrupt requests to the CPU. The INTC has a register to set priority of each interrupt. Interrupt requests are processed according to the priority set in this register by the user.

6.1 Overview

- Simultaneous distribution of interrupt sources to multiple cores
  - A single interrupt source can be distributed to multiple CPU cores.
    (Distribution target cores: CPU1/PCU)
  - Target interrupt sources: Nonmaskable interrupt 1 source, FE level interrupt 1 source, EI level interrupts 21 sources
- Interrupt sources
  - Nonmaskable interrupt:
    External pin NMI interrupt (FENMI) 1 source
  - FE level interrupt:
    ECM interrupt (FEINT) 1 source
  - EI level interrupt (maskable) (EIINT)
    high-speed interrupts (EIINT0 to 31) 21 sources
    • Inter-processor interrupts
    • ECM interrupts
    • External pin IRQ interrupts
    • Software interrupts
    • Constant-cycle timer (OSTM)
    • DMA error interrupts
    Low-speed interrupts (EIINT32 to 511) 462 sources
    • Timer
    • Communication
    • A/D converter
    • DMAC/DTS, etc.
- Priority levels for Interrupt
  Priority levels of IRQ (external interrupts) and maskable interrupts can be set in 512 interrupt control registers according to interrupt requests, up to 16 levels for CPU1 and 8 levels for PCU.
- Detecting methods of external interrupt (NMI/IRQ)
  A method of detecting NMI sources can be selected from two types: rising edge and falling edge.
  A method of detecting IRQ sources can be selected from four types: high level, low level, rising edge, and falling edge.
- Direct branching method or table referencing method is selectable by setting two types of interrupt handler address setting registers.
- Inter-processor interrupts
  High-speed inter-processor interrupts are enabled.
- Software interrupts (SINT)
  Interrupts of desired priority can be generated from the program by setting software interrupt registers.

- Sharing interrupt sources
  The number of exception handler addresses can be decreased by merging multiple interrupt sources.

Interrupts are controlled by the following interrupt controllers:

- INTC1
  CPU1 and PCU have their own interrupt controllers.
  Each PE accesses to the INTC1 register that corresponds to respective PEs.
  INTC1 controls High-speed interrupt and has the following functions:
  - Priority setting
  - Interrupt mask setting
- INTC2
  INTC2 is a common interrupt controller that CPU1 and PCU share.
  INTC2 controls low-speed interrupts and has the following functions.
  - Priority setting
  - Interrupt mask setting
  - Binding setting

6.2 Register Specifications

The INTC has registers listed in tables below. These registers are mainly used to set interrupt priority and to control detection of external interrupt input signals.

6.2.1 Register Configuration

Table 6.1 Interrupt Control

Address	Register Symbol	Register Name	R/W	Value after Reset
FFFE EA00$_H$-FFFE EA3E$_H$ (EIC0 to 31) FFFF B040$_H$-FFFF B3FE$_H$ (EIC32 to 511)	EICn*1	EI level interrupt control register	R/W	008F$_H$*5 808F$_H$*6
FFFE EAF0$_H$ (IMR0) FFFF B404$_H$-FFFF B43C$_H$ (IMR1 to IMR15)	IMRn*2	EI level interrupt mask register	R/W	FFFF FFFF$_H$
FFFE EB00$_H$-FFFE EB7C$_H$ (EIBD0 to 31) FFFF B880$_H$-FFFF BFFC$_H$ (EIBD32 to 511)	EIBDn*3	EI level interrupt bind register	R/W	*4

Note 1. n = 0 to 511
Note 2. n = 0 to 15
Note 3. n = 0 to 511
Note 4. n = 0 to 31: the same value as that of the PEID bit
n = 32 to 511: 0000 0001$_H$
Note 5. When an edge is detected (The PCU register of EICn (n = 0 to 31) is located at 0087$_H$.)
Note 6. When a level signal is detected (The PCU register of EICn (n = 0 to 31) is located at 8087$_H$.)

Among the registers shown in **Table 6.1**, the EIC0 to 31, IMR0, and EIBD0 to 31 are located in INTC1 of the CPU peripheral field included in each CPU. Each register of these only can be accessed from CPU1 or PCU which includes it. Writing is only possible in supervisor mode (PSW.UM = 0).

Of the registers listed in **Table 6.1**, EIC32 to EIC511, IMR1 to IMR15, and EIBD32 to EIBD511 are located in INTC2, the controller for interrupts from peripheral group 0. Writing to these registers is only possible for a PE bound to EIBDn (n = 32 to 511) or in supervisor mode (UM = 0) by CPU1. When writing to IMR1 to IMR15, only the bits corresponding to the conditions described above are overwritten; other bits are not updated.

In the register areas listed in **Table 6.1**, the values of those listed as reserved for the given channel numbers in **Table 6.14, Interrupt Exception Handler and Priority**, must retain their values after a reset.

Table 6.2 External Interrupts, Software Interrupts, and NMI

Address	Register Symbol	Register Name	R/W	Value after Reset
FFC0 0000$_H$	NMICTL	NMI interrupt control register	R/W	00$_H$
FFC0 0010$_H$	EXINTCTL	External interrupt control register	R/W	0000$_H$
FFC0 0014$_H$	EXINTSTR	External interrupt status register	R	00$_H$
FFC0 0018$_H$	EXINTSTC	External interrupt status clear register	W	00$_H$
FFC0 0020$_H$	SINTR0	Software interrupt register 0	R/W	00$_H$
FFC0 0024$_H$	SINTR1	Software interrupt register 1	R/W	00$_H$
FFC0 0028$_H$	SINTR2	Software interrupt register 2	R/W	00$_H$
FFC0 002C$_H$	SINTR3	Software interrupt register 3	R/W	00$_H$

Table 6.3 Interrupt Merge Function

Address	Register Symbol	Register Name	R/W	Value after Reset
FFF9 8000$_H$	PINT0	Peripheral interrupt status register 0	R	0000 0000$_H$
FFF9 8004$_H$	PINT1	Peripheral interrupt status register 1	R	0000 0000$_H$
FFF9 8008$_H$	PINT2	Peripheral interrupt status register 2	R	0000 0000$_H$
FFF9 800C$_H$	PINT3	Peripheral interrupt status register 3	R	0000 0000$_H$
FFF9 8010$_H$	PINT4	Peripheral interrupt status register 4	R	0000 0000$_H$
FFF9 8014$_H$	PINT5	Peripheral interrupt status register 5	R	0000 0000$_H$
FFF9 8018$_H$	PINT6	Peripheral interrupt status register 6	R	0000 0000$_H$
FFF9 801C$_H$	PINT7	Peripheral interrupt status register 7	R	0000 0000$_H$
FFF9 8020$_H$	PINTCLR0	Peripheral interrupt status clear register 0	W	0000 0000$_H$
FFF9 8024$_H$	PINTCLR1	Peripheral interrupt status clear register 1	W	0000 0000$_H$
FFF9 8028$_H$	PINTCLR2	Peripheral interrupt status clear register 2	W	0000 0000$_H$
FFF9 802C$_H$	PINTCLR3	Peripheral interrupt status clear register 3	W	0000 0000$_H$
FFF9 8030$_H$	PINTCLR4	Peripheral interrupt status clear register 4	W	0000 0000$_H$
FFF9 8034$_H$	PINTCLR5	Peripheral interrupt status clear register 5	W	0000 0000$_H$
FFF9 8038$_H$	PINTCLR6	Peripheral interrupt status clear register 6	W	0000 0000$_H$
FFF9 803C$_H$	PINTCLR7	Peripheral interrupt status clear register 7	W	0000 0000$_H$

Table 6.4 ATU-IV Interrupt Control

Address	Register Symbol	Register Name	R/W	Value after Reset
FFF9 8040$_H$	TIMER	Timer interrupt mask enable register	R/W	00$_H$

6.2.2 EIC0 to EIC511 — EI Level Interrupt Control Registers 0 to 511

These registers are used to set interrupt control conditions for each EI level interrupt source, and one register is provided for each source of this type. For each source, see **Table 6.14, Interrupt Exception Handler and Priority**.

CAUTION

If 0 is written to the EIRFn bit immediately after a peripheral module generates the corresponding interrupt request upon detection of an edge (before an interrupt is accepted by the CPU), the request may be lost.

Writing 1 to the EIRFn bit immediately after an interrupt is accepted by the CPU may lead to re-issuing of the request.

This includes the use of bit-manipulation instructions (set1, clr1, and not1) for writing to any of these registers. For bit-manipulation instructions, see also Section 3.5.2, Accesses to Registers by Bit-Manipulation Instructions.

Executing a bit-manipulation instruction to the lower bytes including the EIMKn bit has no effect on the EIRFn bit.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	EICTn	—	—	EIRFn	—	—	—	—	EIMKn	EITBn	—	—	EIPn			
Value after reset	*1	0	0	0	0	0	0	0	1	0	0	0	1	1	1	1
R/W	R	R	R	R/W	R	R	R	R	R/W	R/W	R	R	R/W	R/W	R/W	R/W

Note 1. 0: Edge detection, 1: Level detection

Table 6.5 EIC Register Contents (1/2)

Bit Position	Bit Name	Function
15	EICTn	Interrupt Channel Type The following values are read according to the interrupt input interface. This is a read-only bit. 0: Edge detection 1: Level detection The write value should be the value after reset.
14, 13	—	Reserved These bits are always read as 0. The write value should always be 0.
12	EIRFn	Interrupt Request Flag Operation varies with the interrupt input interface. 0: No interrupt request (Initial value) 1: Interrupt request present • Edge detection This flag is automatically cleared to 0 when an interrupt request of the self-channel is accepted by the CPU core. When the EIRFn bit is set to 1, an EI level maskable interrupt n (EIINTn) is generated in the same way as acceptance of an interrupt request. • Level detection This bit cannot be set or cleared by the software. This is a read-only bit.
11 to 8	—	Reserved These bits are always read as 0. The write value should always be 0.

Table 6.5 EIC Register Contents (2/2)

Bit Position	Bit Name	Function
7	EIMKn	Interrupt Mask While this bit is set to 1, interrupt requests set in the interrupt request flag (EIRFn) are masked to inhibit interrupt requests from the channel to the CPU core. Notification of presence of unprocessed interrupts is not made and the PMEI bit in ICSR is not set from channels for which this bit is set to 1. Even when interrupt processing is disabled by the setting of this bit, an input of interrupt signal is not masked and the interrupt request flag is set. Setting this bit is also reflected in the setting of corresponding bit of the interrupt mask register (IMR). 0: Interrupt processing is enabled. 1: Interrupt processing is disabled. (Initial value)
6	EITBn	Interrupt Vector Method Select 0: Direct branching method based on priority 1: Table referencing method
5, 4	—	Reserved These bits are always read as 0. The write value should always be 0.
3 to 0	EIPn[*1]	These bits specify 16 interrupt priority levels (0: highest priority, 15: lowest priority). If two or more EI level interrupt requests are generated simultaneously, a source with higher priority specified by these bits is selected and is sent to the CPU core. If the priority specified by these bits is equal, a source of smaller channel number is selected as fixed priority.

Note: n = 0 to 511
The values of those listed as reserved for the given channel numbers in **Table 6.14, Interrupt Exception Handler and Priority**, must retain their values after a reset.

Note 1. In the case of the EIC0 to EIC31 registers of the PCU, bits EIPn 2 to EIPn 0 are used to specify eight interrupt priority levels. (Set EIPn 3 to 0).
When interrupts corresponding to EIC32 to EIC511 are bound to the PCU, however, the interrupt priority is judged from the three bits of EIPn 3 to EIPn 1, which the PCU handles as if they were bits EIPn 2 to EIPn 0. (When an interrupt is bound to the PCU, the setting of EIPn 0 in the corresponding EICn is ignored.).

6.2.3 IMR0 to IMR15 — EI Level Interrupt Mask Registers 0 to 15

These registers are aggregation of the EIMK bit in the EIC register. Setting of the corresponding EIMK bit is reflected in each bit in the IMRn register. Furthermore, setting of the IMRn register is reflected in the corresponding EIMK bit.

IMR0

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IMR0H	EIMK31	EIMK30	EIMK29	EIMK28	EIMK27	EIMK26	EIMK25	EIMK24	EIMK23	EIMK22	EIMK21	EIMK20	EIMK19	EIMK18	EIMK17	EIMK16
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IMR0L	EIMK15	EIMK14	EIMK13	EIMK12	EIMK11	EIMK10	EIMK9	EIMK8	EIMK7	EIMK6	EIMK5	EIMK4	EIMK3	EIMK2	EIMK1	EIMK0
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

IMR1

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IMR1H	EIMK63	EIMK62	EIMK61	EIMK60	EIMK59	EIMK58	EIMK57	EIMK56	EIMK55	EIMK54	EIMK53	EIMK52	EIMK51	EIMK50	EIMK49	EIMK48
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IMR1L	EIMK47	EIMK46	EIMK45	EIMK44	EIMK43	EIMK42	EIMK41	EIMK40	EIM39	EIMK38	EIMK37	EIMK36	EIMK35	EIMK34	EIMK33	EIMK32
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

:

:

IMR15

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
IMR15H	EIMK511	EIMK510	EIMK509	EIMK508	EIMK507	EIMK506	EIMK505	EIMK504	EIMK503	EIMK502	EIMK501	EIMK500	EIMK499	EIMK498	EIMK497	EIMK496
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
IMR15L	EIMK495	EIMK494	EIMK493	EIMK492	EIMK491	EIMK490	EIMK489	EIMK488	EIMK487	EIMK486	EIMK485	EIMK484	EIMK483	EIMK482	EIMK481	EIMK480
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note: The value of EIMK bit listed as reserved for the given channel numbers in **Table 6.14, Interrupt Exception Handler and Priority**, must be set to 1.

6.2.4 EIBD0 to EIBD511 — EI Level Interrupt Bind Registers 0 to 511

These registers are provided for each EI level interrupt source to make correspondence between each source and PE. For each source, see **Table 6.14, Interrupt Exception Handler and Priority**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
EIBDnH	—	—	—	—	—	—	—	—	—	—	—	—	—	—	GPID	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
EIBDnL	—	—	—	—	—	—	—	—	—	—	—	—	—	PEID		
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0		*1	
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Table 6.6 EIBD0 to EIBD511 Registers Contents

Bit Position	Bit Name	Function
31 to 18	—	Reserved The write value should always be 0.
17, 16	GPID	These bits are provided only in EIBD32 to EIBD511. Set either of the following values for these bits according to the PEID setting. 00: When CPU1 is selected by PEID as a bind destination 01: When PCU is selected by PEID as a bind destination These bits are reserved for EIBD0 to EIBD31. The write value should always be 0. These bits are always read as 0.
15 to 3	—	Reserved The write value should always be 0.
2 to 0	PEID	These bits specify an interrupt bind (request) destination. However, these bits are fixed in EIBD0 to EIBD31 and cannot be modified. 001: Interrupts are bound to CPU1. 011: Interrupts are bound to the PCU. To execute interrupt of the corresponding source, be sure to set either value above.

Note: The values of those listed as reserved for the given channel numbers in **Table 6.14, Interrupt Exception Handler and Priority**, must retain their values after a reset.

Note 1. EIBD32 to EIBD511: 001

NOTE

Changing the corresponding EIBDn register during the processing of an EIINT request is prohibited.

6.2.5 NMICTL — NMI Interrupt Control Register

NMICTL is an 8-bit register to set the detection mode of falling edge, or rising edge for the NMI interrupt input.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	NMIS	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 6.7 NMICTL Register Contents

Bit Position	Bit Name	Function
7 to 2	—	Reserved
1, 0	NMIS	NMI Interrupt Detecting Method Select These bits select an NMI interrupt signal detecting method from low level, high level, falling edge, and rising edge. 0 0: An interrupt request is detected by the NMI input falling edge. 0 1: An interrupt request is detected by the NMI input rising edge. 1 0: Setting prohibited 1 1: Setting prohibited

6.2.6 EXINTCTL — External Interrupt Control Register

EXINTCTL is a 16-bit register to specify a low level, high level, falling edge, or rising edge detection mode individually for external interrupt input pins IRQ7 to IRQ0.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IRQ7S		IRQ6S		IRQ5S		IRQ4S		IRQ3S		IRQ2S		IRQ1S		IRQ1S	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 6.8 EXINTCTL Register Contents

Bit Position	Bit Name	Function
15, 14	IRQ7S	External Interrupt Detecting Method Select These bits select an IRQ7 to IRQ0 interrupt signal detecting method from low level, high level, falling edge, and rising edge. 0 0: An interrupt request is detected by the IRQn input low level.*1 0 1: An interrupt request is detected by the IRQn input high level.*1 1 0: An interrupt request is detected by the IRQn input falling edge. 1 1: An interrupt request is detected by the IRQn input rising edge.
13, 12	IRQ6S	
11, 10	IRQ5S	
9, 8	IRQ4S	
7, 6	IRQ3S	
5, 4	IRQ2S	
3, 2	IRQ1S	
1, 0	IRQ0S	

Note 1. When level detection is selected, the active level should be retained until an interrupt request is accepted.

6.2.7 EXINTSTR — External Interrupt Status Register

EXINTSTR is an 8-bit register that indicates interrupt requests of external interrupt input pins IRQ7 to IRQ0. When edge detection is specified for IRQ7 to IRQ0 interrupts, interrupt requests retained in the EXINTSTC register can be cleared.

Bit	7	6	5	4	3	2	1	0
	IRQ7F	IRQ6F	IRQ5F	IRQ4F	IRQ3F	IRQ2F	IRQ1F	IRQ0F
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 6.9 EXINTSTR Register Contents

Bit Position	Bit Name	Function
7	IRQ7F	External Interrupt Request These bits indicate IRQ7 to IRQ0 interrupt request status. • When level detection is selected 0: No IRQn interrupt request is present. 1: IRQn interrupt requests are present. [Clearing condition] The IRQn input is not the level corresponding to IRQn1S and IRQn0S in EXINTCTL. [Setting condition] The IRQn input is the level corresponding to IRQn1S and IRQn0S in EXINTCTL • When edge detection is selected 0: No IRQn interrupt request has been detected. 1: IRQn interrupt requests have been detected. [Clearing condition] Writing 1 to the IRQnC bit in EXINTSTC [Setting condition] An edge corresponding to IRQn1S and IRQn0S in EXINTCTL is generated.
6	IRQ6F	
5	IRQ5F	
4	IRQ4F	
3	IRQ3F	
2	IRQ2F	
1	IRQ1F	
0	IRQ0F	

Note 1. n = 0 to 7

6.2.8 EXINTSTC — External Interrupt Status Clear Register

EXINTSTC is an 8-bit register to clear IRQnF in EXINTSTR in case edge detection is selected as an IRQn detecting method. Writing 1 to IRQnC clears the corresponding IRQnF bit in EXINTSTR.

Bit	7	6	5	4	3	2	1	0
	IRQ7C	IRQ6C	IRQ5C	IRQ4C	IRQ3C	IRQ2C	IRQ1C	IRQ0C
Value after reset	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W

Table 6.10 EXINTSTC Register Contents

Bit Position	Bit Name	Function
7	IRQ7C	External Interrupt Request Clear These bits clear the interrupt request status in case edge detection method is selected for detecting IRQ7 to IRQ0 interrupts. • When level detection is selected These bits have no function. • When edge detection is selected Writing 1 clears the corresponding IRQnF bit in the EXINTSTR register.
6	IRQ6C	
5	IRQ5C	
4	IRQ4C	
3	IRQ3C	
2	IRQ2C	
1	IRQ1C	
0	IRQ0C	

Note 1. n = 0 to 7

6.2.9 SINTR0 to SINTR3 — Software Interrupt Registers

SINTR0 to SINTR3 are 8-bit registers to control software interrupts 0 to 3 (SINT0 to SINT3).

Writing 01_H to these registers increments the value of a counter; writing 00_H decrements it. When the value of the counter for any of these registers is one or more, the corresponding interrupt from among software interrupts 0 to 3 (SINT0 to SINT3) is generated. When reading the value from any of these registers, it is the current value of the counter.

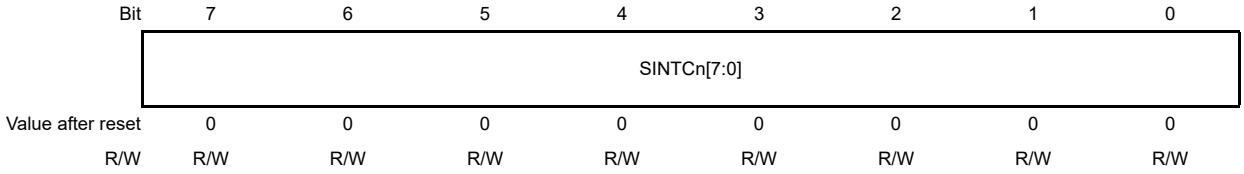


Table 6.11 SINTR0 to SINTR3 Register Contents

Bit Position	Bit Name	Function
7 to 0	SINTCn[7:0]	Software Interrupt Request This bit generates a software interrupt. [Reading operation] The number of SINTn interrupt request counts is read out. [Writing operation] Writing 01_H: increments the counter.*1 Writing 00_H: decrements the counter.*2

Note 1. When 01_H is written to the register while the value of the counter is FF_H, the counter is not incremented and its value remains FF_H

Note 2. When 00_H is written to the register while the value of the counter is 00_H, the counter is not decremented and its value remains 00_H.

6.2.10 PINT0 to PINT7, PINTCLR0 to PINTCLR7 — Peripheral Interrupt Status Registers and Peripheral Interrupt Status Clear Registers

The DTS transfer completion interrupt and transfer count match interrupt are merged in 32-channel units. PINT0 to PINT7 contain the interrupt status flags to indicate the originating channels for these interrupts. When multiple interrupt sources are generated within the same register among PINT0 to PINT7, only bit 1 on the lower-order side of the bits is set.

Writing the value read from the peripheral interrupt status register (PINT0 to PINT7) to the interrupt clear register (PINTCLR0 to PINTCLR7) of the same channel in the interrupt handler clears interrupts.

PINT n + x (n = 0 to 3, x = 0)

Bit	31	30	29	28	27	26	25	24
	INTDTS [31+32*n]	INTDTS [30+32*n]	INTDTS [29+32*n]	INTDTS [28+32*n]	INTDTS [27+32*n]	INTDTS [26+32*n]	INTDTS [25+32*n]	INTDTS [24+32*n]
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Bit	23	22	21	20	19	18	17	16
	INTDTS [23+32*n]	INTDTS [22+32*n]	INTDTS [21+32*n]	INTDTS [20+32*n]	INTDTS [19+32*n]	INTDTS [18+32*n]	INTDTS [17+32*n]	INTDTS [16+32*n]
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8
	INTDTS [15+32*n]	INTDTS [14+32*n]	INTDTS [13+32*n]	INTDTS [12+32*n]	INTDTS [11+32*n]	INTDTS [10+32*n]	INTDTS [9+32*n]	INTDTS [8+32*n]
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Bit	7	6	5	4	3	2	1	0
	INTDTS [7+32*n]	INTDTS [6+32*n]	INTDTS [5+32*n]	INTDTS [4+32*n]	INTDTS [3+32*n]	INTDTS [2+32*n]	INTDTS [1+32*n]	INTDTS [0+32*n]
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

PINT n + x (n = 0 to 3, x = 4)

Bit	31	30	29	28	27	26	25	24
	INTCTDTS [31+32*n]	INTCTDTS [30+32*n]	INTCTDTS [29+32*n]	INTCTDTS [28+32*n]	INTCTDTS [27+32*n]	INTCTDTS [26+32*n]	INTCTDTS [25+32*n]	INTCTDTS [24+32*]
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Bit	23	22	21	20	19	18	17	16
	INTCTDTS [23+32*n]	INTCTDTS [22+32*n]	INTCTDTS [21+32*n]	INTCTDTS [20+32*n]	INTCTDTS [19+32*n]	INTCTDTS [18+32*n]	INTCTDTS [17+32*n]	INTCTDTS [16+32*n]
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8
	INTCTDTS [15+32*n]	INTCTDTS [14+32*n]	INTCTDTS [13+32*n]	INTCTDTS [12+32*n]	INTCTDTS [11+32*n]	INTCTDTS [10+32*n]	INTCTDTS [9+32*n]	INTCTDTS [8+32*n]
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Bit	7	6	5	4	3	2	1	0
	INTCTDTS [7+32*n]	INTCTDTS [6+32*n]	INTCTDTS [5+32*n]	INTCTDTS [4+32*n]	INTCTDTS [3+32*n]	INTCTDTS [2+32*n]	INTCTDTS [1+32*n]	INTCTDTS [0+32*n]
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

PINTCLR n + x (n = 0 to 3, x = 0)

Bit	31	30	29	28	27	26	25	24
	INTCLR [31+32*n]	INTCLR [30+32*n]	INTCLR [29+32*n]	INTCLR [28+32*n]	INTCLR [27+32*n]	INTCLR [26+32*n]	INTCLR [25+32*n]	INTCLR [24+32*n]
Value after reset	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W

Bit	23	22	21	20	19	18	17	16
	INTCLR [23+32*n]	INTCLR [22+32*n]	INTCLR [21+32*n]	INTCLR [20+32*n]	INTCLR [19+32*n]	INTCLR [18+32*n]	INTCLR [17+32*n]	INTCLR [16+32*n]
Value after reset	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8
	INTCLR [15+32*n]	INTCLR [14+32*n]	INTCLR [13+32*n]	INTCLR [12+32*n]	INTCLR [11+32*n]	INTCLR [10+32*n]	INTCLR [9+32*n]	INTCLR [8+32*n]
Value after reset	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W

Bit	7	6	5	4	3	2	1	0
	INTCLR [7+32*n]	INTCLR [6+32*n]	INTCLR [5+32*n]	INTCLR [4+32*n]	INTCLR [3+32*n]	INTCLR [2+32*n]	INTCLR [1+32*n]	INTCLR [0+32*n]
Value after reset	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W

PINTCLR n + x (n = 0 to 3, x = 4)

Bit	31	30	29	28	27	26	25	24
	INTCTCLR [31+32*n]	INTCTCLR [30+32*n]	INTCTCLR [29+32*n]	INTCTCLR [28+32*n]	INTCTCLR [27+32*n]	INTCTCLR [26+32*n]	INTCTCLR [25+32*n]	INTCTCLR [24+32*]
Value after reset	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W

Bit	23	22	21	20	19	18	17	16
	INTCTCLR [23+32*n]	INTCTCLR [22+32*n]	INTCTCLR [21+32*n]	INTCTCLR [20+32*n]	INTCTCLR [19+32*n]	INTCTCLR [18+32*n]	INTCTCLR [17+32*n]	INTCTCLR [16+32*n]
Value after reset	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8
	INTCTCLR [15+32*]	INTCTCLR [14+32*n]	INTCTCLR [13+32*n]	INTCTCLR [12+32*n]	INTCTCLR [11+32*n]	INTCTCLR [10+32*n]	INTCTCLR [9+32*n]	INTCTCLR [8+32*n]
Value after reset	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W

Bit	7	6	5	4	3	2	1	0
	INTCTCLR [7+32*n]	INTCTCLR [6+32*n]	INTCTCLR [5+32*n]	INTCTCLR [4+32*n]	INTCTCLR [3+32*n]	INTCTCLR [2+32*n]	INTCTCLR [1+32*n]	INTCTCLR [0+32*n]
Value after reset	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W

PINT0

Bit Position	Bit Name	Function
31 to 0	INTDTS[31:0]	DTS transfer completion interrupt status on ch31 to ch0

PINT1

Bit Position	Bit Name	Function
31 to 0	INTDTS[63:32]	DTS transfer completion interrupt status on ch63 to ch32

PINT2

Bit Position	Bit Name	Function
31 to 0	INTDTS[95:64]	DTS transfer completion interrupt status on ch95 to ch64

PINT3

Bit Position	Bit Name	Function
31 to 0	INTDTS[127:96]	DTS transfer completion interrupt status on ch127 to ch96

PINT4

Bit Position	Bit Name	Function
31 to 0	INTCTDTS[31:0]	DTS transfer count match interrupt status on ch31 to ch0

PINT5

Bit Position	Bit Name	Function
31 to 0	INTCTDTS [63:32]	DTS transfer count match interrupt status on ch63 to ch32

PINT6

Bit Position	Bit Name	Function
31 to 0	INTCTDTS [95:64]	DTS transfer count match interrupt status on ch95 to ch64

PINT7

Bit Position	Bit Name	Function
31 to 0	INTCTDTS [127:96]	DTS transfer count match interrupt status on ch127 to ch96

PINTCLR0

Bit Position	Bit Name	Function
31 to 0	INTCLR[31:0]	These bits clear the DTS transfer completion interrupt status on ch31 to ch0. These bits write the read value of the PINT0 in the interrupt handler.

PINTCLR1

Bit Position	Bit Name	Function
31 to 0	INTCLR[63:32]	These bits clear the DTS transfer completion interrupt status on ch63 to ch32. These bits write the read value of the PINT1 in the interrupt handler.

PINTCLR2

Bit Position	Bit Name	Function
31 to 0	INTCLR[95:64]	These bits clear the DTS transfer completion interrupt status on ch95 to ch64. These bits write the read value of the PINT2 in the interrupt handler.

PINTCLR3

Bit Position	Bit Name	Function
31 to 0	INTCLR[127:96]	These bits clear the DTS transfer completion interrupt status on ch127 to ch96. These bits write the read value of the PINT3 in the interrupt handler.

PINTCLR4

Bit Position	Bit Name	Function
31 to 0	INTCTCLR[31:0]	These bits clear the DTS transfer count match interrupt status on ch31 to ch0. These bits write the read value of the PINT4 in the interrupt handler.

PINTCLR5

Bit Position	Bit Name	Function
31 to 0	INTCTCLR [63:32]	These bits clear the DTS transfer count match interrupt status on ch63 to ch32. These bits write the read value of the PINT5 in the interrupt handler.

PINTCLR6

Bit Position	Bit Name	Function
31 to 0	INTCTCLR [95:64]	These bits clear the DTS transfer count match interrupt status on ch95 to ch64. These bits write the read value of the PINT6 in the interrupt handler.

PINTCLR7

Bit Position	Bit Name	Function
31 to 0	INTCTCLR [127:96]	These bits clear the DTS transfer count match interrupt status on ch127 to ch96. These bits write the read value of the PINT7 in the interrupt handler.

6.2.11 TIMER — Timer Interrupt Mask Enable Register

TIMER is an 8-bit readable/writable register to enable or disable interrupt requests of subblocks 4 to 6 of the ATU-IV timer D. **Table 6.13, TIMER Register Target Interrupts** lists applicable interrupts.

Since these interrupts are merged with the TSG2 interrupts and input to the INTC, set the corresponding IMEn bits to 1 when using the TSG2. When subblock 6 of the timer D is used only as an A/D converter or a trigger output to the DFE, set the IME0 bit to 1. Do not modify these bits while subblock 4, 5, or 6 of the ATU-IV timer D is operating.

TIMER can be read and written on a byte basis.

TIMER is initialized to 00_H by a reset.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	IME2	IME1	IME0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W

Table 6.12 TIMER Register Contents

Bit Position	Bit Name	Function
7 to 3	—	Nothing is allocated to these bits. These bits should always be 0.
2	IME2	This bit enables or disables interrupt requests of subblock 4 of ATU-IV timer D. 0: Interrupt requests are enabled. 1: Interrupt requests are disabled.
1	IME1	This bit enables or disables interrupt requests of subblock 5 of ATU-IV timer D. 0: Interrupt requests are enabled. 1: Interrupt requests are disabled.
0	IME0	This bit enables or disables interrupt requests of subblock 6 of ATU-IV timer D. 0: Interrupt requests are enabled. 1: Interrupt requests are disabled.

The following lists the target interrupts.

Table 6.13 TIMER Register Target Interrupts

TIMER Register Target Bit	ATU-IV Timer D Subblock	Interrupt Channel	Interrupt Name
IME2	4	180	DCNTD40 down-counter underflow interrupt
		181	DCNTD41 down-counter underflow interrupt
		182	DCNTD42 down-counter underflow interrupt
		183	DCNTD43 down-counter underflow interrupt
IME1	5	184	OCR1D50 compare match interrupt
		185	OCR2D50 compare match interrupt
		186	OCR1D51 compare match interrupt
		187	OCR2D51 compare match interrupt
		188	OCR1D52 compare match interrupt
		189	OCR2D52 compare match interrupt
		190	OCR1D53 compare match interrupt
		191	OCR2D53 compare match interrupt
		192	TCNT1D5 overflow interrupt
		193	TCNT2D5 overflow interrupt
		194	DCNTD50 down-counter underflow interrupt
		195	DCNTD51 down-counter underflow interrupt
		196	DCNTD52 down-counter underflow interrupt
		197	DCNTD53 down-counter underflow interrupt
IME0	6	198	OCR1D60 compare match interrupt
		199	OCR2D60 compare match interrupt
		200	OCR1D61 compare match interrupt
		201	OCR2D61 compare match interrupt
		202	OCR1D62 compare match interrupt
		203	OCR2D62 compare match interrupt
		204	OCR1D63 compare match interrupt
		205	OCR2D63 compare match interrupt
		206	TCNT1D6 overflow interrupt
		207	TCNT2D6 overflow interrupt

6.3 Interrupt Sources

Interrupt sources are grouped into five: external interrupt (NMI/IRQ), ECM interrupt, inter-processor interrupt, software interrupt, and peripheral module interrupt.

6.3.1 NMI Interrupts

NMI interrupts are external interrupts input from the NMI pin. Falling edge or rising edge is selectable by the setting of the NMI Interrupt Detecting Method Select bits (NMIS) in the NMI interrupt control register (NMICTL).

NMI interrupts are accepted at the high priority even when other FE level interrupt has been generated. NMI interrupts cannot be masked regardless of the state of PSW.NP in the CPU system register. NMI interrupts are non-maskable interrupts from which return or recovery is not possible.

6.3.2 IRQ Interrupts

IRQ interrupts are input from IRQ7 to IRQ0 pins. For IRQ interrupts, low level, high level, falling edge, or rising edge is selectable for each pin by the setting of the External Interrupt Detecting Method Select bits (IRQ7S to IRQ0S) in the external interrupt control register (EXINTCTL). Furthermore, interrupt priority levels can be set for each source in interrupt control registers, 16 levels for CPU1 and 8 levels for PCU.

When low level is selected as an IRQ interrupt detecting method, an interrupt request signal is sent to the INTC while IRQ7 to IRQ0 pins are low level. When IRQ7 to IRQ0 pins become high level, no interrupt request signal is sent to the INTC. The active level should be retained until an interrupt request is accepted. Interrupt requests can be checked by reading IRQ Interrupt Request bits (IRQ7F to IRQ0F) in the external interrupt status register (EXINTSTR).

When edge detection is selected as an IRQ interrupt detecting method, an interrupt request is detected by a change in IRQ7 to IRQ0 pins, and an interrupt request signal is sent to the INTC. Reading bits EXINTSTR.IRQ7F to EXINTSTR.IRQ0F checks whether the corresponding IRQ interrupt request is detected. In addition, the IRQ interrupt request can be cleared by writing 1 to the bit corresponding to EXINTSTC when an edge is detected.

When returning from the IRQ interrupt exception handler, clear the external interrupt status register (EXINTSTR) to prevent re-acceptance by mistake, and then issue an instruction to return from interrupt.

6.3.3 ECM Interrupts

The Error Control Module (ECM) generates ECM interrupt requests by merging error interrupts from multiple sources. For details, see **Section 30, Error Control Module (ECM)**.

6.3.4 Inter-Processor Interrupts

Four registers (IPIR_CHn) for conveying interrupts between PEs are provided.

IPIR_CH0 to IPIR_CH3 are assigned to CH0 to CH3 of user interrupt (EIINT). An interrupt for

specific PEs (including own PE) can be requested by manipulating bits corresponding to respective PEs. Interrupt priority levels can be set for each source in interrupt control registers, 16 levels for CPU1 and 8 levels for PCU.

6.3.5 Software Interrupts

The software interrupts (SINT) are generated by the setting of the SINTR0 to SINTR3 registers. Multiple interrupt requests can be queued.

Interrupt priority levels can be set for each source in interrupt control registers, 16 levels for CPU1 and 8 levels for PCU.

6.3.6 On-Chip Peripheral Module Interrupts

On-chip peripheral module interrupts occur in on-chip peripheral modules listed below.

- Code flash/Data flash
- Serial communication interface 3 (SCI3)
- OS timer (OSTM)
- Window watchdog timer (WDTA)
- Advanced timer unit IV (ATU-IV)
- Motor control timer (TSG2)
- Autonomous pulse control adapter (APA)
- AD converter (ADCB)
- AD addition function (ASF)
- Delta-sigma AD converter (DS-ADC)
- Serial communication interface H (CSIH)
- CAN interface (RS-CAN)
- FlexRay
- LIN master interface (RLIN2)
- Renesas high-speed bus (RHSB)
- Direct memory access controller (DMAC, DTS)
- Digital filter engine (DFE)

Since different interrupt vectors are assigned to each source, the interrupt exception handler need not decide sources. Interrupt priority levels can be set for each source in interrupt control registers, 16 levels for CPU1 and 8 levels for PCU.

6.4 Interrupt Exception Handler and Priority Operations

Table 6.14 lists interrupt sources, source codes, exception handler offset addresses, and interrupt priority.

There are two specifications for exception handler addresses: (1) standard specifications where exception handler addresses are determined by the PSW.EBV bit in the CPU core, RBASE register, and EBASE register and (2) extended specifications where exception handler addresses for interrupts are specified individually for each channel.

In the standard specifications, an offset address is added to the base address (RBASE register /EBASE register) in the CPU core to generate an exception handler address. The following two methods are provided for giving an interrupt offset address. For channels other than the interrupt channel, the specified offset address is given.

- An offset address is determined within a range of $+100_H$ to $+1F0_H$ according to the priority level (0 to 15) specified for each channel, regardless of interrupt channels (Note 1 in **Table 6.14**).
- Every offset address is $+ 100_H$ regardless of the priority level. This is a function to reduce the memory occupation size of the exception handler (Note 2 in **Table 6.14**).

In the extended specifications, a table for reading exception handler addresses for each interrupt channel is provided. The handler address is extracted by referencing the table. The table reference position is obtained by the following calculation formula (Note 3 in **Table 6.14**). The INTBP register exists in the CPU core.

Exception handler address read position = INTBP + channel number * 4-byte

For RH850 G3M and G3K exceptions, refer to “RH850G3M User's Manual: Software and RH850G3K User's Manual: Software”.

For system error exceptions, see **Section 3, CPU System**.

Priority can be set for each channel. If specified priority levels are equal, an interrupt source of smaller channel number is selected as fixed priority.

CAUTION

Be sure to place a SYNCP instruction at the start of handlers for the FENMI, FEINT, EIINT (direct vector method) and SYSERR and FPI exceptions.

For details, refer to the *RH850G3M User’s Manual: Software*.

Table 6.14 Interrupt Exception Handler and Priority (1/13)

Functional Module	Interrupt Source Name	Level Interrupt Note	Number of Interrupt Channel (EIINT)	Source Code	Offset Address: Direct Branch RINT = 0 Note1	Offset Address: Direct Branch RINT = 1 Note2	Offset Address: Table Reference Note3	Interrupt Priority (Value after reset)	Default Priority
Non-maskable interrupt	NMI interrupt		(FENMI)	$E0_H$	$+0E0_H$	$+0E0_H$	—		High
FE level interrupt	Error control module (ECM) FE level interrupt		(FEINT)	$F0_H$	$+0F0_H$	$+0F0_H$	—		
Inter-processor interrupt	IPIR_CH0		0	1000	The offset address is determined within a range of $+100_H$ to $+1F0_H$ according to the priority, regardless of interrupt channels.	The offset address is always $+100_H$ regardless of priority.	$+000_H$	0 to 15(15)	
	IPIR_CH1		1	1001			$+004_H$	0 to 15(15)	
	IPIR_CH2		2	1002			$+008_H$	0 to 15(15)	
	IPIR_CH3		3	1003			$+00C_H$	0 to 15(15)	
	Reserved		4	1004					
	Reserved		5	1005					
	Reserved		6	1006					
	Reserved		7	1007					
Error control module	Error control module (ECM) maskable interrupt		8	1008			$+020_H$	0 to 15(15)	
IRQ (External interrupts)	IRQ0 interrupt	√	9	1009			$+024_H$	0 to 15(15)	
	IRQ1 interrupt	√	10	100A			$+028_H$	0 to 15(15)	
	IRQ2 interrupt	√	11	100B			$+02C_H$	0 to 15(15)	
	IRQ3 interrupt	√	12	100C			$+030_H$	0 to 15(15)	
	IRQ4 interrupt	√	13	100D			$+034_H$	0 to 15(15)	
	IRQ5 interrupt	√	14	100E			$+038_H$	0 to 15(15)	
	IRQ6 interrupt	√	15	100F			$+03C_H$	0 to 15(15)	
	IRQ7 interrupt	√	16	1010			$+040_H$	0 to 15(15)	
SINT (software interrupts)	SINT0 interrupt	√	17	1011			$+044_H$	0 to 15(15)	
	SINT1 interrupt	√	18	1012			$+048_H$	0 to 15(15)	
	SINT2 interrupt	√	19	1013			$+04C_H$	0 to 15(15)	
	SINT3 interrupt	√	20	1014			$+050_H$	0 to 15(15)	
	Reserved		21	1015					
	Reserved		22	1016					
	Reserved		23	1017					
	Reserved		24	1018					
OSTM	OSTM0 interrupt (OSTM0TINT)		25	1019			$+064_H$	0 to 15(15)	
	OSTM1 interrupt (OSTM1TINT)		26	101A			$+068_H$	0 to 15(15)	
	OSTM2 interrupt (OSTM2TINT)		27	101B			$+06C_H$	0 to 15(15)	
	Reserved		28	101C					
DMA	DMA transfer error (DMAERR)		29	101D			$+074_H$	0 to 15(15)	
	Reserved		30	101E					
	Reserved		31	101F					
	Reserved		32	1020					
Code Flash/Data Flash	Flash sequencer processing end interrupt*1		33	1021			$+084_H$	0 to 15(15)	
	Reserved		34	1022					
	Reserved		35	1023					
	Reserved		36	1024					
	Reserved		37	1025					
DMAC	Ch0 transfer end interrupt / ch0 transfer count match interrupt		38	1026			$+098_H$	0 to 15(15)	
	Ch1 transfer end interrupt / ch1 transfer count match interrupt		39	1027			$+09C_H$	0 to 15(15)	
	Ch2 transfer end interrupt / ch2 transfer count match interrupt		40	1028			$+0A0_H$	0 to 15(15)	
	Ch3 transfer end interrupt / ch3 transfer count match interrupt		41	1029			$+0A4_H$	0 to 15(15)	

Table 6.14 Interrupt Exception Handler and Priority (2/13)

Functional Module	Interrupt Source Name	Level Interrupt Note	Number of Interrupt Channel (EIINT)	Source Code	Offset Address: Direct Branch RINT = 0 Note1	Offset Address: Direct Branch RINT = 1 Note2	Table Reference Note3	Interrupt Priority (Value after reset)	Default Priority
DMAC	Ch4 transfer end interrupt / ch4 transfer count match interrupt		42	102A	The offset address is determined within a range of $+100_H$ to $+1F0_H$ according to the priority, regardless of interrupt channels.	The offset address is always $+100_H$ regardless of priority.	$+0A8_H$	0 to 15(15)	High
	Ch5 transfer end interrupt / ch5 transfer count match interrupt		43	102B			$+0AC_H$	0 to 15(15)	
	Ch6 transfer end interrupt / ch6 transfer count match interrupt		44	102C			$+0B0_H$	0 to 15(15)	
	Ch7 transfer end interrupt / ch7 transfer count match interrupt		45	102D			$+0B4_H$	0 to 15(15)	
	Ch8 transfer end interrupt / ch8 transfer count match interrupt		46	102E			$+0B8_H$	0 to 15(15)	
	Ch9 transfer end interrupt / ch9 transfer count match interrupt		47	102F			$+0BC_H$	0 to 15(15)	
	Ch10 transfer end interrupt / ch10 transfer count match interrupt		48	1030			$+0C0_H$	0 to 15(15)	
	Ch11 transfer end interrupt / ch11 transfer count match interrupt		49	1031			$+0C4_H$	0 to 15(15)	
	Ch12 transfer end interrupt / ch12 transfer count match interrupt		50	1032			$+0C8_H$	0 to 15(15)	
	Ch13 transfer end interrupt / ch13 transfer count match interrupt		51	1033			$+0CC_H$	0 to 15(15)	
	Ch14 transfer end interrupt / ch14 transfer count match interrupt		52	1034			$+0D0_H$	0 to 15(15)	
	Ch15 transfer end interrupt / ch15 transfer count match interrupt		53	1035			$+0D4_H$	0 to 15(15)	
WDTA	WDTA0TIT interval timer interrupt (75% interrupt)		54	1036			$+0D8_H$	0 to 15(15)	
	WDTA1TIT interval timer interrupt (75% interrupt)		55	1037			$+0DC_H$	0 to 15(15)	
	Reserved		56	1038					
ATU-IV timer A	ICRA0 input capture interrupt		57	1039			$+0E4_H$	0 to 15(15)	
	ICRA1 input capture interrupt		58	103A			$+0E8_H$	0 to 15(15)	
	ICRA2 input capture interrupt		59	103B			$+0EC_H$	0 to 15(15)	
	ICRA3 input capture interrupt		60	103C			$+0F0_H$	0 to 15(15)	
	ICRA4 input capture interrupt		61	103D			$+0F4_H$	0 to 15(15)	
	ICRA5 input capture interrupt		62	103E			$+0F8_H$	0 to 15(15)	
	ICRA6 input capture interrupt		63	103F			$+0FC_H$	0 to 15(15)	
	TCNTA overflow interrupt		64	1040			$+100_H$	0 to 15(15)	
ATU-IV timer B	OCRB0 compare match interrupt		65	1041			$+104_H$	0 to 15(15)	
	OCRB1 compare match interrupt		66	1042			$+108_H$	0 to 15(15)	
	OCRB6 compare match interrupt		67	1043			$+10C_H$	0 to 15(15)	
	OCRB10 compare match interrupt		68	1044			$+110_H$	0 to 15(15)	
	OCRB11 compare match interrupt		69	1045			$+114_H$	0 to 15(15)	
	OCRB12 compare match interrupt		70	1046			$+118_H$	0 to 15(15)	
	Comparison between TCNTB6M and ICRB6 condition match interrupt		71	1047			$+11C_H$	0 to 15(15)	
	AND/OR condition of CMFB6 and CMFB6M condition match interrupt		72	1048			$+120_H$	0 to 15(15)	
	ICRB0 input capture interrupt		73	1049			$+124_H$	0 to 15(15)	
ATU-IV timer C	GRC00 input capture/compare match interrupt/ OCRC000 compare match interrupt		74	104A			$+128_H$	0 to 15(15)	
	GRC01 input capture/compare match interrupt/ OCRC001 compare match interrupt		75	104B			$+12C_H$	0 to 15(15)	
	GRC02 input capture/compare match interrupt/ OCRC002 compare match interrupt		76	104C			$+130_H$	0 to 15(15)	
	GRC03 input capture/compare match interrupt/ OCRC003 compare match interrupt		77	104D			$+134_H$	0 to 15(15)	
	TCNTC0 overflow interrupt		78	104E			$+138_H$	0 to 15(15)	

Table 6.14 Interrupt Exception Handler and Priority (3/13)

Functional Module	Interrupt Source Name	Level Interrupt Note	Number of Interrupt Channel (EIINT)	Source Code	Offset Address: Direct Branch RINT = 0 Note1	Offset Address: Direct Branch RINT = 1 Note2	Table Reference Note3	Interrupt Priority (Value after reset)	Default Priority
ATU-IV timer C	GRC10 input capture/compare match interrupt/ OCRC10 compare match interrupt		79	104F	The offset address is determined within a range of $+100_H$ to $+1F0_H$ according to the priority, regardless of interrupt channels.	The offset address is always $+100_H$ regardless of priority.	$+13C_H$	0 to 15(15)	High
	GRC11 input capture/compare match interrupt/ OCRC11 compare match interrupt		80	1050			$+140_H$	0 to 15(15)	
	GRC12 input capture/compare match interrupt/ OCRC12 compare match interrupt		81	1051			$+144_H$	0 to 15(15)	
	GRC13 input capture/compare match interrupt/ OCRC13 compare match interrupt		82	1052			$+148_H$	0 to 15(15)	
	TCNTC1 overflow interrupt		83	1053			$+14C_H$	0 to 15(15)	
	GRC20 input capture/compare match interrupt/ OCRC20 compare match interrupt		84	1054			$+150_H$	0 to 15(15)	
	GRC21 input capture/compare match interrupt/ OCRC21 compare match interrupt		85	1055			$+154_H$	0 to 15(15)	
	GRC22 input capture/compare match interrupt/ OCRC22 compare match interrupt		86	1056			$+158_H$	0 to 15(15)	
	GRC23 input capture/compare match interrupt/ OCRC23 compare match interrupt		87	1057			$+15C_H$	0 to 15(15)	
	TCNTC2 overflow interrupt		88	1058			$+160_H$	0 to 15(15)	
	GRC30 input capture/compare match interrupt/ OCRC30 compare match interrupt		89	1059			$+164_H$	0 to 15(15)	
	GRC31 input capture/compare match interrupt/ OCRC31 compare match interrupt		90	105A			$+168_H$	0 to 15(15)	
	GRC32 input capture/compare match interrupt/ OCRC32 compare match interrupt		91	105B			$+16C_H$	0 to 15(15)	
	GRC33 input capture/compare match interrupt/ OCRC33 compare match interrupt		92	105C			$+170_H$	0 to 15(15)	
	TCNTC3 overflow interrupt		93	105D			$+174_H$	0 to 15(15)	
	GRC40 input capture/compare match interrupt/ OCRC40 compare match interrupt		94	105E			$+178_H$	0 to 15(15)	
	GRC41 input capture/compare match interrupt/ OCRC41 compare match interrupt		95	105F			$+17C_H$	0 to 15(15)	
	GRC42 input capture/compare match interrupt/ OCRC42 compare match interrupt		96	1060			$+180_H$	0 to 15(15)	
	GRC43 input capture/compare match interrupt/ OCRC43 compare match interrupt		97	1061			$+184_H$	0 to 15(15)	
	TCNTC4 overflow interrupt		98	1062			$+188_H$	0 to 15(15)	
	GRC50 input capture/compare match interrupt/ OCRC50 compare match interrupt		99	1063			$+18C_H$	0 to 15(15)	
	GRC51input capture/compare match interrupt/ OCRC51 compare match interrupt		100	1064			$+190_H$	0 to 15(15)	
	GRC52 input capture/compare match interrupt/ OCRC52 compare match interrupt		101	1065			$+194_H$	0 to 15(15)	
	GRC53 input capture/compare match interrupt/ OCRC53 compare match interrupt		102	1066			$+198_H$	0 to 15(15)	
	TCNTC5 overflow interrupt		103	1067			$+19C_H$	0 to 15(15)	
	GRC60 input capture/compare match interrupt/ OCRC60 compare match interrupt		104	1068			$+1A0_H$	0 to 15(15)	
	GRC61 input capture/compare match interrupt/ OCRC61 compare match interrupt		105	1069			$+1A4_H$	0 to 15(15)	
	GRC62 input capture/compare match interrupt/ OCRC62 compare match interrupt		106	106A			$+1A8_H$	0 to 15(15)	
	GRC63 input capture/compare match interrupt/ OCRC63 compare match interrupt		107	106B			$+1AC_H$	0 to 15(15)	
	TCNTC6 overflow interrupt		108	106C			$+1B0_H$	0 to 15(15)	
	GRC70 input capture/compare match interrupt/ OCRC70 compare match interrupt		109	106D			$+1B4_H$	0 to 15(15)	

Table 6.14 Interrupt Exception Handler and Priority (4/13)

Functional Module	Interrupt Source Name	Level Interrupt Note	Number of Interrupt Channel (EIINT)	Source Code	Offset Address: Direct Branch RINT = 0 Note1	Offset Address: Direct Branch RINT = 1 Note2	Table Reference Note3	Interrupt Priority (Value after reset)	Default Priority
ATU-IV timer C	GRC71 input capture/compare match interrupt/ OCRC71 compare match interrupt		110	106E	The offset address is determined within a range of $+100_H$ to $+1F0_H$ according to the priority, regardless of interrupt channels.	The offset address is always $+100_H$ regardless of priority.	$+1B8_H$	0 to 15(15)	High
	GRC72 input capture/compare match interrupt/ OCRC72 compare match interrupt		111	106F			$+1BC_H$	0 to 15(15)	
	GRC73 input capture/compare match interrupt/ OCRC73 compare match interrupt		112	1070			$+1C0_H$	0 to 15(15)	
	TCNTC7 overflow interrupt		113	1071			$+1C4_H$	0 to 15(15)	
ATU-IV timer D	OCR1D00 compare match interrupt		114	1072			$+1C8_H$	0 to 15(15)	
	OCR2D00 compare match interrupt		115	1073			$+1CC_H$	0 to 15(15)	
	OCR1D01 compare match interrupt		116	1074			$+1D0_H$	0 to 15(15)	
	OCR2D01 compare match interrupt		117	1075			$+1D4_H$	0 to 15(15)	
	OCR1D02 compare match interrupt		118	1076			$+1D8_H$	0 to 15(15)	
	OCR2D02 compare match interrupt		119	1077			$+1DC_H$	0 to 15(15)	
	OCR1D03 compare match interrupt		120	1078			$+1E0_H$	0 to 15(15)	
	OCR2D03 compare match interrupt		121	1079			$+1E4_H$	0 to 15(15)	
	TCNT1D0 overflow interrupt		122	107A			$+1E8_H$	0 to 15(15)	
	TCNT2D0 overflow interrupt		123	107B			$+1EC_H$	0 to 15(15)	
	DCNTD00 down-counter underflow interrupt		124	107C			$+1F0_H$	0 to 15(15)	
	DCNTD01 down-counter underflow interrupt		125	107D			$+1F4_H$	0 to 15(15)	
	DCNTD02 down-counter underflow interrupt		126	107E			$+1F8_H$	0 to 15(15)	
	DCNTD03 down-counter underflow interrupt		127	107F			$+1FC_H$	0 to 15(15)	
	OCR1D10 compare match interrupt		128	1080			$+200_H$	0 to 15(15)	
	OCR2D10 compare match interrupt		129	1081			$+204_H$	0 to 15(15)	
	OCR1D11 compare match interrupt		130	1082			$+208_H$	0 to 15(15)	
	OCR2D11 compare match interrupt		131	1083			$+20C_H$	0 to 15(15)	
	OCR1D12 compare match interrupt		132	1084			$+210_H$	0 to 15(15)	
	OCR2D12 compare match interrupt		133	1085			$+214_H$	0 to 15(15)	
	OCR1D13 compare match interrupt		134	1086			$+218_H$	0 to 15(15)	
	OCR2D13 compare match interrupt		135	1087			$+21C_H$	0 to 15(15)	
	TCNT1D1 overflow interrupt		136	1088			$+220_H$	0 to 15(15)	
	TCNT2D1 overflow interrupt		137	1089			$+224_H$	0 to 15(15)	
	DCNTD10 down-counter underflow interrupt		138	108A			$+228_H$	0 to 15(15)	
	DCNTD11 down-counter underflow interrupt		139	108B			$+22C_H$	0 to 15(15)	
	DCNTD12 down-counter underflow interrupt		140	108C			$+230_H$	0 to 15(15)	
	DCNTD13 down-counter underflow interrupt		141	108D			$+234_H$	0 to 15(15)	
	OCR1D20 compare match interrupt		142	108E			$+238_H$	0 to 15(15)	
	OCR2D20 compare match interrupt		143	108F			$+23C_H$	0 to 15(15)	
	OCR1D21 compare match interrupt		144	1090			$+240_H$	0 to 15(15)	
	OCR2D21 compare match interrupt		145	1091			$+244_H$	0 to 15(15)	
	OCR1D22 compare match interrupt		146	1092			$+248_H$	0 to 15(15)	
	OCR2D22 compare match interrupt		147	1093			$+24C_H$	0 to 15(15)	
	OCR1D23 compare match interrupt		148	1094			$+250_H$	0 to 15(15)	
	OCR2D23 compare match interrupt		149	1095			$+254_H$	0 to 15(15)	
	TCNT1D2 overflow interrupt		150	1096			$+258_H$	0 to 15(15)	
	TCNT2D2 overflow interrupt		151	1097			$+25C_H$	0 to 15(15)	
	DCNTD20 down-counter underflow interrupt		152	1098			$+260_H$	0 to 15(15)	
	DCNTD21 down-counter underflow interrupt		153	1099			$+264_H$	0 to 15(15)	
	DCNTD22 down-counter underflow interrupt		154	109A			$+268_H$	0 to 15(15)	
	DCNTD23 down-counter underflow interrupt		155	109B			$+26C_H$	0 to 15(15)	
	OCR1D30 compare match interrupt		156	109C			$+270_H$	0 to 15(15)	

Table 6.14 Interrupt Exception Handler and Priority (5/13)

Functional Module	Interrupt Source Name	Level Interrupt Note	Number of Interrupt Channel (EIINT)	Source Code	Offset Address: Direct Branch RINT = 0 Note1	Offset Address: Direct Branch RINT = 1 Note2	Table Reference Note3	Interrupt Priority (Value after reset)	Default Priority
ATU-IV timer D	OCR2D30 compare match interrupt		157	109D	The offset address is determined within a range of $+100_H$ to $+1F0_H$ according to the priority, regardless of interrupt channels.	The offset address is always $+100_H$ regardless of priority.	$+274_H$	0 to 15(15)	High
	OCR1D31 compare match interrupt		158	109E			$+278_H$	0 to 15(15)	
	OCR2D31 compare match interrupt		159	109F			$+27C_H$	0 to 15(15)	
	OCR1D32 compare match interrupt		160	10A0			$+280_H$	0 to 15(15)	
	OCR2D32 compare match interrupt		161	10A1			$+284_H$	0 to 15(15)	
	OCR1D33 compare match interrupt		162	10A2			$+288_H$	0 to 15(15)	
	OCR2D33 compare match interrupt		163	10A3			$+28C_H$	0 to 15(15)	
	TCNT1D3 overflow interrupt		164	10A4			$+290_H$	0 to 15(15)	
	TCNT2D3 overflow interrupt		165	10A5			$+294_H$	0 to 15(15)	
	DCNTD30 down-counter underflow interrupt		166	10A6			$+298_H$	0 to 15(15)	
	DCNTD31 down-counter underflow interrupt		167	10A7			$+29C_H$	0 to 15(15)	
	DCNTD32 down-counter underflow interrupt		168	10A8			$+2A0_H$	0 to 15(15)	
	DCNTD33 down-counter underflow interrupt		169	10A9			$+2A4_H$	0 to 15(15)	
	OCR1D40 compare match interrupt		170	10AA			$+2A8_H$	0 to 15(15)	
	OCR2D40 compare match interrupt		171	10AB			$+2AC_H$	0 to 15(15)	
	OCR1D41 compare match interrupt		172	10AC			$+2B0_H$	0 to 15(15)	
	OCR2D41 compare match interrupt		173	10AD			$+2B4_H$	0 to 15(15)	
	OCR1D42 compare match interrupt		174	10AE			$+2B8_H$	0 to 15(15)	
	OCR2D42 compare match interrupt		175	10AF			$+2BC_H$	0 to 15(15)	
	OCR1D43 compare match interrupt		176	10B0			$+2C0_H$	0 to 15(15)	
	OCR2D43 compare match interrupt		177	10B1			$+2C4_H$	0 to 15(15)	
	TCNT1D4 overflow interrupt		178	10B2			$+2C8_H$	0 to 15(15)	
	TCNT2D4 overflow interrupt		179	10B3			$+2CC_H$	0 to 15(15)	
ATU-IV timer D	DCNTD40 down-counter underflow interrupt		180	10B4			$+2D0_H$	0 to 15(15)	
TSG2_0	TSG20 compare match interrupt 1 (INTTSG20I1)								
ATU-IV timer D	DCNTD41 down-counter underflow interrupt		181	10B5			$+2D4_H$	0 to 15(15)	
TSG2_0	TSG20 compare match interrupt 2 (INTTSG20I2)								
ATU-IV timer D	DCNTD42 down-counter underflow interrupt		182	10B6			$+2D8_H$	0 to 15(15)	
TSG2_0	TSG20 compare match interrupt 3 (INTTSG20I3)								
ATU-IV timer D	DCNTD43 down-counter underflow interrupt		183	10B7			$+2DC_H$	0 to 15(15)	
TSG2_0	TSG20 compare match interrupt 4 (INTTSG20I4)								
ATU-IV timer D	OCR1D50 compare match interrupt		184	10B8			$+2E0_H$	0 to 15(15)	
TSG2_0	TSG20 compare match interrupt 5 (INTTSG20I5)								
ATU-IV timer D	OCR2D50 compare match interrupt		185	10B9			$+2E4_H$	0 to 15(15)	
TSG2_0	TSG20 compare match interrupt 6 (INTTSG20I6)								
ATU-IV timer D	OCR1D51 compare match interrupt		186	10BA			$+2E8_H$	0 to 15(15)	
TSG2_0	TSG20 compare match interrupt 7 (INTTSG20I7)								
ATU-IV timer D	OCR2D51 compare match interrupt		187	10BB			$+2EC_H$	0 to 15(15)	
TSG2_0	TSG20 compare match interrupt 8 (INTTSG20I8)								
ATU-IV timer D	OCR1D52 compare match interrupt		188	10BC			$+2F0_H$	0 to 15(15)	
TSG2_0	TSG20 compare match interrupt 9 (INTTSG20I9)								
ATU-IV timer D	OCR2D52 compare match interrupt		189	10BD			$+2F4_H$	0 to 15(15)	
TSG2_0	TSG20 compare match interrupt 10 (INTTSG20I10)								
ATU-IV timer D	OCR1D53 compare match interrupt		190	10BE			$+2F8_H$	0 to 15(15)	
TSG2_0	TSG20 compare match interrupt 11 (INTTSG20I11)								

Table 6.14 Interrupt Exception Handler and Priority (6/13)

Functional Module	Interrupt Source Name	Level Interrupt Note	Number of Interrupt Channel (EIINT)	Source Code	Offset Address: Direct Branch RINT = 0 Note1	Offset Address: Direct Branch RINT = 1 Note2	Table Reference Note3	Interrupt Priority (Value after reset)	Default Priority
ATU-IV timer D	OCR2D53 compare match interrupt		191	10BF	The offset address is determined within a range of $+100_H$ to $+1F0_H$ according to the priority, regardless of interrupt channels.	The offset address is always $+100_H$ regardless of priority.	$+2FC_H$	0 to 15(15)	High
TSG2_0	TSG20 compare match interrupt 12 (INTTSG20I12)								
ATU-IV timer D	TCNT1D5 overflow interrupt		192	10C0			$+300_H$	0 to 15(15)	
TSG2_0	TSG20 error interrupt (INTTSG20IER)								
ATU-IV timer D	TCNT2D5 overflow interrupt		193	10C1			$+304_H$	0 to 15(15)	
TSG2_0	TSG20 warning interrupt (INTTSG20IWN)								
ATU-IV timer D	DCNTD50 down-counter underflow interrupt		194	10C2			$+308_H$	0 to 15(15)	
TSG2_1	TSG21 compare match interrupt 1 (INTTSG21I1)								
ATU-IV timer D	DCNTD51 down-counter underflow interrupt		195	10C3			$+30C_H$	0 to 15(15)	
TSG2_1	TSG21 compare match interrupt 2 (INTTSG21I2)								
ATU-IV timer D	DCNTD52 down-counter underflow interrupt		196	10C4			$+310_H$	0 to 15(15)	
TSG2_1	TSG21 compare match interrupt 3 (INTTSG21I3)								
ATU-IV timer D	DCNTD53 down-counter underflow interrupt		197	10C5			$+314_H$	0 to 15(15)	
TSG2_1	TSG21 compare match interrupt 4 (INTTSG21I4)								
ATU-IV timer D	OCR1D60 compare match interrupt		198	10C6			$+318_H$	0 to 15(15)	
TSG2_1	TSG21 compare match interrupt 5 (INTTSG21I5)								
ATU-IV timer D	OCR2D60 compare match interrupt		199	10C7			$+31C_H$	0 to 15(15)	
TSG2_1	TSG21 compare match interrupt 6 (INTTSG21I6)								
ATU-IV timer D	OCR1D61 compare match interrupt		200	10C8			$+320_H$	0 to 15(15)	
TSG2_1	TSG21 compare match interrupt 7 (INTTSG21I7)								
ATU-IV timer D	OCR2D61 compare match interrupt		201	10C9			$+324_H$	0 to 15(15)	
TSG2_1	TSG21 compare match interrupt 8 (INTTSG21I8)								
ATU-IV timer D	OCR1D62 compare match interrupt		202	10CA			$+328_H$	0 to 15(15)	
TSG2_1	TSG21 compare match interrupt 9 (INTTSG21I9)								
ATU-IV timer D	OCR2D62 compare match interrupt		203	10CB			$+32C_H$	0 to 15(15)	
TSG2_1	TSG21 compare match interrupt 10 (INTTSG21I10)								
ATU-IV timer D	OCR1D63 compare match interrupt		204	10CC			$+330_H$	0 to 15(15)	
TSG2_1	TSG21 compare match interrupt 11 (INTTSG21I11)								
ATU-IV timer D	OCR2D63 compare match interrupt		205	10CD			$+334_H$	0 to 15(15)	
TSG2_1	TSG21 compare match interrupt 12 (INTTSG21I12)								
ATU-IV timer D	TCNT1D6 overflow interrupt		206	10CE			$+338_H$	0 to 15(15)	
TSG2_1	TSG21 error interrupt (INTTSG21IER)								
ATU-IV timer D	TCNT2D6 overflow interrupt		207	10CF			$+33C_H$	0 to 15(15)	
TSG2_1	TSG21 warning interrupt (INTTSG21IWM)								
ATU-IV timer D	DCNTD60 down-counter underflow interrupt		208	10D0			$+340_H$	0 to 15(15)	
	DCNTD61 down-counter underflow interrupt		209	10D1			$+344_H$	0 to 15(15)	
	DCNTD62 down-counter underflow interrupt		210	10D2			$+348_H$	0 to 15(15)	
	DCNTD63 down-counter underflow interrupt		211	10D3			$+34C_H$	0 to 15(15)	
	OCR1D70 compare match interrupt		212	10D4			$+350_H$	0 to 15(15)	
	OCR2D70 compare match interrupt/ DCNTD70 down-counter underflow interrupt		213	10D5			$+354_H$	0 to 15(15)	
	OCR1D71compare match interrupt		214	10D6			$+358_H$	0 to 15(15)	
	OCR2D71 compare match interrupt/ DCNTD71 down-counter underflow interrupt		215	10D7			$+35C_H$	0 to 15(15)	
	OCR1D72 compare match interrupt		216	10D8			$+360_H$	0 to 15(15)	
	OCR2D72 compare match interrupt/ DCNTD72 down-counter underflow interrupt		217	10D9			$+364_H$	0 to 15(15)	

Table 6.14 Interrupt Exception Handler and Priority (7/13)

Functional Module	Interrupt Source Name	Level Interrupt Note	Number of Interrupt Channel (EIINT)	Source Code	Offset Address: Direct Branch RINT = 0 Note1	Offset Address: Direct Branch RINT = 1 Note2	Table Reference Note3	Interrupt Priority (Value after reset)	Default Priority
ATU-IV timer D	OCR1D73 compare match interrupt		218	10DA	The offset address is determined within a range of $+100_H$ to $+1F0_H$ according to the priority, regardless of interrupt channels.	The offset address is always $+100_H$ regardless of priority.	$+368_H$	0 to 15(15)	High
	OCR2D73 compare match interrupt/ DCNTD73 down-counter underflow interrupt		219	10DB			$+36C_H$	0 to 15(15)	
	TCNT1D7 overflow interrupt		220	10DC			$+370_H$	0 to 15(15)	
	TCNT2D7 overflow interrupt		221	10DD			$+374_H$	0 to 15(15)	
	OCR1D80 compare match interrupt		222	10DE			$+378_H$	0 to 15(15)	
	OCR2D80 compare match interrupt/ DCNTD80 down-counter underflow interrupt		223	10DF			$+37C_H$	0 to 15(15)	
	OCR1D81 compare match interrupt		224	10E0			$+380_H$	0 to 15(15)	
	OCR2D81 compare match interrupt/ DCNTD81 down-counter underflow interrupt		225	10E1			$+384_H$	0 to 15(15)	
	OCR1D82 compare match interrupt		226	10E2			$+388_H$	0 to 15(15)	
	OCR2D82 compare match interrupt/ DCNTD82 down-counter underflow interrupt		227	10E3			$+38C_H$	0 to 15(15)	
	OCR1D83 compare match interrupt		228	10E4			$+390_H$	0 to 15(15)	
	OCR2D83 compare match interrupt/ DCNTD83 down-counter underflow interrupt		229	10E5			$+394_H$	0 to 15(15)	
	TCNT1D8 overflow interrupt		230	10E6			$+398_H$	0 to 15(15)	
	TCNT2D8 overflow interrupt		231	10E7			$+39C_H$	0 to 15(15)	
	OCR1D90 compare match interrupt		232	10E8			$+3A0_H$	0 to 15(15)	
	OCR2D90 compare match interrupt/ DCNTD90 down-counter underflow interrupt		233	10E9			$+3A4_H$	0 to 15(15)	
	OCR1D91 compare match interrupt		234	10EA			$+3A8_H$	0 to 15(15)	
	OCR2D91 compare match interrupt/ DCNTD91 down-counter underflow interrupt		235	10EB			$+3AC_H$	0 to 15(15)	
	OCR1D92 compare match interrupt		236	10EC			$+3B0_H$	0 to 15(15)	
	OCR2D92 compare match interrupt/ DCNTD92 down-counter underflow interrupt		237	10ED			$+3B4_H$	0 to 15(15)	
	OCR1D93 compare match interrupt		238	10EE			$+3B8_H$	0 to 15(15)	
	OCR2D93 compare match interrupt/ DCNTD93 down-counter underflow interrupt		239	10EF			$+3BC_H$	0 to 15(15)	
	TCNT1D9 overflow interrupt		240	10F0			$+3C0_H$	0 to 15(15)	
	TCNT2D9 overflow interrupt		241	10F1			$+3C4_H$	0 to 15(15)	
ATU-IV timer E	CYLRE00 cycle compare match interrupt / DTRE00 duty compare match interrupt 00	√	242	10F2			$+3C8_H$	0 to 15(15)	
	CYLRE01 cycle compare match interrupt / DTRE01 duty compare match interrupt 01	√	243	10F3			$+3CC_H$	0 to 15(15)	
	CYLRE02 cycle compare match interrupt / DTRE02 duty compare match interrupt 02	√	244	10F4			$+3D0_H$	0 to 15(15)	
	CYLRE03 cycle compare match interrupt / DTRE03 duty compare match interrupt 03	√	245	10F5			$+3D4_H$	0 to 15(15)	
	CYLRE10 cycle compare match interrupt / DTRE10 duty compare match interrupt 10	√	246	10F6			$+3D8_H$	0 to 15(15)	
	CYLRE11 cycle compare match interrupt / DTRE11 duty compare match interrupt 11	√	247	10F7			$+3DC_H$	0 to 15(15)	
	CYLRE12 cycle compare match interrupt / DTRE12 duty compare match interrupt 12	√	248	10F8			$+3E0_H$	0 to 15(15)	
	CYLRE13 cycle compare match interrupt / DTRE13 duty compare match interrupt 13	√	249	10F9			$+3E4_H$	0 to 15(15)	
	CYLRE20 cycle compare match interrupt / DTRE20 duty compare match interrupt 20	√	250	10FA			$+3E8_H$	0 to 15(15)	
	CYLRE21 cycle compare match interrupt / DTRE21 duty compare match interrupt 21	√	251	10FB			$+3EC_H$	0 to 15(15)	
	CYLRE22 cycle compare match interrupt / DTRE22 duty compare match interrupt 22	√	252	10FC			$+3F0_H$	0 to 15(15)	
	CYLRE23 cycle compare match interrupt / DTRE23 duty compare match interrupt 23	√	253	10FD			$+3F4_H$	0 to 15(15)	
	CYLRE30 cycle compare match interrupt / DTRE30 duty compare match interrupt 30	√	254	10FE			$+3F8_H$	0 to 15(15)	

Table 6.14 Interrupt Exception Handler and Priority (8/13)

Functional Module	Interrupt Source Name	Level Interrupt Note	Number of Interrupt Channel (EIINT)	Source Code	Offset Address: Direct Branch RINT = 0 Note1	Offset Address: Direct Branch RINT = 1 Note2	Table Reference Note3	Interrupt Priority (Value after reset)	Default Priority
ATU-IV timer E	CYLRE31 cycle compare match interrupt / DTRE31 duty compare match interrupt 31	√	255	10FF	The offset address is determined within a range of $+100_H$ to $+1F0_H$ according to the priority, regardless of interrupt channels.	The offset address is always $+100_H$ regardless of priority.	$+3FC_H$	0 to 15(15)	High
	CYLRE32 cycle compare match interrupt / DTRE32 duty compare match interrupt 32	√	256	1100			$+400_H$	0 to 15(15)	
	CYLRE33 cycle compare match interrupt / DTRE33 duty compare match interrupt 33	√	257	1101			$+404_H$	0 to 15(15)	
	CYLRE40 cycle compare match interrupt / DTRE40 duty compare match interrupt 40	√	258	1102			$+408_H$	0 to 15(15)	
	CYLRE41 cycle compare match interrupt / DTRE41 duty compare match interrupt 41	√	259	1103			$+40C_H$	0 to 15(15)	
	CYLRE42 cycle compare match interrupt / DTRE42 duty compare match interrupt 42	√	260	1104			$+410_H$	0 to 15(15)	
	CYLRE43 cycle compare match interrupt / DTRE43 duty compare match interrupt 43	√	261	1105			$+414_H$	0 to 15(15)	
	CYLRE50 cycle compare match interrupt / DTRE50 duty compare match interrupt 50	√	262	1106			$+418_H$	0 to 15(15)	
	CYLRE51 cycle compare match interrupt / DTRE51 duty compare match interrupt 51	√	263	1107			$+41C_H$	0 to 15(15)	
	CYLRE52 cycle compare match interrupt / DTRE52 duty compare match interrupt 52	√	264	1108			$+420_H$	0 to 15(15)	
	CYLRE53 cycle compare match interrupt / DTRE53 duty compare match interrupt 53	√	265	1109			$+424_H$	0 to 15(15)	
	CYLRE60 cycle compare match interrupt / DTRE60 duty compare match interrupt 60	√	266	110A			$+428_H$	0 to 15(15)	
	CYLRE61 cycle compare match interrupt / DTRE61 duty compare match interrupt 61	√	267	110B			$+42C_H$	0 to 15(15)	
	CYLRE62 cycle compare match interrupt / DTRE62 duty compare match interrupt 62	√	268	110C			$+430_H$	0 to 15(15)	
	CYLRE63 cycle compare match interrupt / DTRE63 duty compare match interrupt 63	√	269	110D			$+434_H$	0 to 15(15)	
ATU-IV timer F	Timer F0 overflow interrupt	√	270	110E			$+438_H$	0 to 15(15)	
	Timer F1 overflow interrupt	√	271	110F			$+43C_H$	0 to 15(15)	
	Timer F2 overflow interrupt	√	272	1110			$+440_H$	0 to 15(15)	
	Timer F3 overflow interrupt	√	273	1111			$+444_H$	0 to 15(15)	
	Timer F4 overflow interrupt	√	274	1112			$+448_H$	0 to 15(15)	
	Timer F5 overflow interrupt	√	275	1113			$+44C_H$	0 to 15(15)	
	Timer F6 overflow interrupt	√	276	1114			$+450_H$	0 to 15(15)	
	Timer F7 overflow interrupt	√	277	1115			$+454_H$	0 to 15(15)	
	Timer F8 overflow interrupt	√	278	1116			$+458_H$	0 to 15(15)	
	Timer F9 overflow interrupt	√	279	1117			$+45C_H$	0 to 15(15)	
	Timer F10 overflow interrupt	√	280	1118			$+460_H$	0 to 15(15)	
	Timer F11 overflow interrupt	√	281	1119			$+464_H$	0 to 15(15)	
	Timer F12 overflow interrupt	√	282	111A			$+468_H$	0 to 15(15)	
	Timer F13 overflow interrupt	√	283	111B			$+46C_H$	0 to 15(15)	
	Timer F14 overflow interrupt	√	284	111C			$+470_H$	0 to 15(15)	
	Timer F15 overflow interrupt	√	285	111D			$+474_H$	0 to 15(15)	
	Timer F0 input capture interrupt		286	111E			$+478_H$	0 to 15(15)	
	Timer F1 input capture interrupt		287	111F			$+47C_H$	0 to 15(15)	
	Timer F2 input capture interrupt		288	1120			$+480_H$	0 to 15(15)	
	Timer F3 input capture interrupt		289	1121			$+484_H$	0 to 15(15)	
	Timer F4 input capture interrupt		290	1122			$+488_H$	0 to 15(15)	

Table 6.14 Interrupt Exception Handler and Priority (9/13)

Functional Module	Interrupt Source Name	Level Interrupt Note	Number of Interrupt Channel (EIINT)	Source Code	Offset Address: Direct Branch RINT = 0 Note1	Offset Address: Direct Branch RINT = 1 Note2	Table Reference Note3	Interrupt Priority (Value after reset)	Default Priority
ATU-IV timer F	Timer F5 input capture interrupt		291	1123	The offset address is determined within a range of $+100_H$ to $+1F0_H$ according to the priority, regardless of interrupt channels.	The offset address is always $+100_H$ regardless of priority.	$+48C_H$	0 to 15(15)	High
	Timer F6 input capture interrupt		292	1124			$+490_H$	0 to 15(15)	
	Timer F7 input capture interrupt		293	1125			$+494_H$	0 to 15(15)	
	Timer F8 input capture interrupt		294	1126			$+498_H$	0 to 15(15)	
	Timer F9 input capture interrupt		295	1127			$+49C_H$	0 to 15(15)	
	Timer F10 input capture interrupt		296	1128			$+4A0_H$	0 to 15(15)	
	Timer F11 input capture interrupt		297	1129			$+4A4_H$	0 to 15(15)	
	Timer F12 input capture interrupt		298	112A			$+4A8_H$	0 to 15(15)	
	Timer F13 input capture interrupt		299	112B			$+4AC_H$	0 to 15(15)	
	Timer F14 input capture interrupt		300	112C			$+4B0_H$	0 to 15(15)	
	Timer F15 input capture interrupt		301	112D			$+4B4_H$	0 to 15(15)	
ATU-IV timer G	OCRG0 compare match interrupt		302	112E			$+4B8_H$	0 to 15(15)	
	OCRG1 compare match interrupt		303	112F			$+4BC_H$	0 to 15(15)	
	OCRG2 compare match interrupt		304	1130			$+4C0_H$	0 to 15(15)	
	OCRG3 compare match interrupt		305	1131			$+4C4_H$	0 to 15(15)	
	OCRG4 compare match interrupt		306	1132			$+4C8_H$	0 to 15(15)	
	OCRG5 compare match interrupt		307	1133			$+4CC_H$	0 to 15(15)	
	OCRG6 compare match interrupt		308	1134			$+4D0_H$	0 to 15(15)	
	OCRG7 compare match interrupt		309	1135			$+4D4_H$	0 to 15(15)	
	OCRG8 compare match interrupt		310	1136			$+4D8_H$	0 to 15(15)	
	OCRG9 compare match interrupt		311	1137			$+4DC_H$	0 to 15(15)	
ATU-IV timer H	OCR1H compare match interrupt		312	1138			$+4E0_H$	0 to 15(15)	
ATU-IV timer J	FIFOJ0 FIFO data full interrupt		313	1139			$+4E4_H$	0 to 15(15)	
	FIFOJ1 FIFO data full interrupt		314	113A			$+4E8_H$	0 to 15(15)	
	FIFOJ2 FIFO data full interrupt		315	113B			$+4EC_H$	0 to 15(15)	
	FIFOJ3 FIFO data full interrupt		316	113C			$+4F0_H$	0 to 15(15)	
	FIFOJ4 FIFO data full interrupt		317	113D			$+4F4_H$	0 to 15(15)	
	FIFOJ5 FIFO data full interrupt		318	113E			$+4F8_H$	0 to 15(15)	
	Reserved		319	113F					
	Reserved		320	1140					
ATU-IV timer J	TCNTJ0 overflow interrupt		321	1141			$+504_H$	0 to 15(15)	
	TCNTJ1 overflow interrupt		322	1142			$+508_H$	0 to 15(15)	
	TCNTJ2 overflow interrupt		323	1143			$+50C_H$	0 to 15(15)	
	TCNTJ3 overflow interrupt		324	1144			$+510_H$	0 to 15(15)	
	TCNTJ4 overflow interrupt		325	1145			$+514_H$	0 to 15(15)	
	TCNTJ5 overflow interrupt		326	1146			$+518_H$	0 to 15(15)	
	Reserved		327	1147					
	Reserved		328	1148					
ATU-IV timer J	FIFOJ0 FIFO data overflow interrupt		329	1149			$+524_H$	0 to 15(15)	
	FIFOJ1 FIFO data overflow interrupt		330	114A			$+528_H$	0 to 15(15)	
	FIFOJ2 FIFO data overflow interrupt		331	114B			$+52C_H$	0 to 15(15)	
	FIFOJ3 FIFO data overflow interrupt		332	114C			$+530_H$	0 to 15(15)	
	FIFOJ4 FIFO data overflow interrupt		333	114D			$+534_H$	0 to 15(15)	
	FIFOJ5 FIFO data overflow interrupt		334	114E			$+538_H$	0 to 15(15)	
	Reserved		335	114F					
	Reserved		336	1150					
TSG2_0	TSG20 compare match interrupt 0 (INTTSG20I0)		337	1151			$+544_H$	0 to 15(15)	
	TSG20 peak interrupt (INTTSG20IPEK)		338	1152			$+548_H$	0 to 15(15)	
	TSG2 trough interrupt (INTTSG20IVLY)		339	1153			$+54C_H$	0 to 15(15)	

Table 6.14 Interrupt Exception Handler and Priority (10/13)

Functional Module	Interrupt Source Name	Level Interrupt Note	Number of Interrupt Channel (EIINT)	Source Code	Offset Address: Direct Branch RINT = 0 Note1	Offset Address: Direct Branch RINT = 1 Note2	Offset Address: Table Reference Note3	Interrupt Priority (Value after reset)	Default Priority
TSG2_1	TSG21 compare match interrupt 0 (INTTSG21I0)		340	1154	The offset address is determined within a range of $+100_H$ to $+1F0_H$ according to the priority, regardless of interrupt channels.	The offset address is always $+100_H$ regardless of priority.	$+550_H$	0 to 15(15)	High
	TSG21 peak interrupt (INTTSG21IPEK)		341	1155			$+554_H$	0 to 15(15)	
	TSG21 trough interrupt (INTTSG21IVLY)		342	1156			$+558_H$	0 to 15(15)	
APA	PWM output ch.0 status change interrupt		343	1157			$+55C_H$	0 to 15(15)	
	PWM output ch.1 status change interrupt		344	1158			$+560_H$	0 to 15(15)	
	PWM output ch.2 status change interrupt		345	1159			$+564_H$	0 to 15(15)	
	PWM output ch.3 status change interrupt		346	115A			$+568_H$	0 to 15(15)	
	PWM output ch.4 status change interrupt		347	115B			$+56C_H$	0 to 15(15)	
	PWM output ch.5 status change interrupt		348	115C			$+570_H$	0 to 15(15)	
	PWM output ch.6 status change interrupt		349	115D			$+574_H$	0 to 15(15)	
	PWM output ch.7 status change interrupt		350	115E			$+578_H$	0 to 15(15)	
	PWM output ch.8 status change interrupt		351	115F			$+57C_H$	0 to 15(15)	
	PWM output ch.9 status change interrupt		352	1160			$+580_H$	0 to 15(15)	
	PWM output ch.10 status change interrupt		353	1161			$+584_H$	0 to 15(15)	
	PWM output ch.11 status change interrupt		354	1162			$+588_H$	0 to 15(15)	
	PWM output ch.12 status change interrupt		355	1163			$+58C_H$	0 to 15(15)	
	PWM output ch.13 status change interrupt		356	1164			$+590_H$	0 to 15(15)	
	PWM output ch.14 status change interrupt		357	1165			$+594_H$	0 to 15(15)	
	PWM output ch.15 status change interrupt		358	1166			$+598_H$	0 to 15(15)	
DFE	CH0 output data interrupt / CH0 condition match interrupt		359	1167			$+59C_H$	0 to 15(15)	
	CH1 output data interrupt / CH1 condition match interrupt		360	1168			$+5A0_H$	0 to 15(15)	
	CH2 output data interrupt / CH2 condition match interrupt		361	1169			$+5A4_H$	0 to 15(15)	
	CH3 output data interrupt / CH3 condition match interrupt		362	116A			$+5A8_H$	0 to 15(15)	
	CH4 output data interrupt / CH4 condition match interrupt		363	116B			$+5AC_H$	0 to 15(15)	
	CH5 output data interrupt / CH5 condition match interrupt		364	116C			$+5B0_H$	0 to 15(15)	
	CH6 output data interrupt / CH6 condition match interrupt		365	116D			$+5B4_H$	0 to 15(15)	
	CH7 output data interrupt / CH7 condition match interrupt		366	116E			$+5B8_H$	0 to 15(15)	
	CH8 output data interrupt / CH8 condition match interrupt		367	116F			$+5BC_H$	0 to 15(15)	
	CH9 output data interrupt / CH9 condition match interrupt		368	1170			$+5C0_H$	0 to 15(15)	
	CH10 output data interrupt		369	1171			$+5C4_H$	0 to 15(15)	
	CH11 output data interrupt		370	1172			$+5C8_H$	0 to 15(15)	
	CH12 output data interrupt		371	1173			$+5CC_H$	0 to 15(15)	
	CH13 output data interrupt		372	1174			$+5D0_H$	0 to 15(15)	
	CH14 output data interrupt		373	1175			$+5D4_H$	0 to 15(15)	
	CH15 output data interrupt		374	1176			$+5D8_H$	0 to 15(15)	
ADCB	ADI00 ADC0 scan group 0 end interrupt		375	1177			$+5DC_H$	0 to 15(15)	
	ADI00 ADC0 scan group 1 end interrupt		376	1178			$+5E0_H$	0 to 15(15)	
	ADI00 ADC0 scan group 2 end interrupt		377	1179			$+5E4_H$	0 to 15(15)	
	ADI00 ADC0 scan group 3 end interrupt		378	117A			$+5E8_H$	0 to 15(15)	
	ADI04 ADC0 scan group 4 end interrupt		379	117B			$+5EC_H$	0 to 15(15)	
	ADI10 ADC1 scan group 0 end interrupt		380	117C			$+5F0_H$	0 to 15(15)	
	ADI11 ADC1 scan group 1 end interrupt		381	117D			$+5F4_H$	0 to 15(15)	
	ADI12 ADC1 scan group 2 end interrupt		382	117E			$+5F8_H$	0 to 15(15)	

Table 6.14 Interrupt Exception Handler and Priority (11/13)

Functional Module	Interrupt Source Name	Level Interrupt Note	Number of Interrupt Channel (EIINT)	Source Code	Offset Address: Direct Branch RINT = 0 Note1	Offset Address: Direct Branch RINT = 1 Note2	Table Reference Note3	Interrupt Priority (Value after reset)	Default Priority
ADCB	ADI13 ADC1 scan group 3 end interrupt		383	117F	The offset address is determined within a range of $+100_H$ to $+1F0_H$ according to the priority, regardless of interrupt channels.	The offset address is always $+100_H$ regardless of priority.	$+5FC_H$	0 to 15(15)	High
	ADI14 ADC1 scan group 4 end interrupt		384	1180			$+600_H$	0 to 15(15)	
	ADC0 MPX interrupt (ADMPXI0)		385	1181			$+604_H$	0 to 15(15)	
	ADC1 MPX interrupt (ADMPXI1)		386	1182			$+608_H$	0 to 15(15)	
	ADE0 ADC0AD error interrupt		387	1183			$+60C_H$	0 to 15(15)	
	ADE1 ADC1AD error interrupt		388	1184			$+610_H$	0 to 15(15)	
ASF (AD addition function)	ASI0 ch0 integration end interrupt		389	1185			$+614_H$	0 to 15(15)	
	ASI1 ch1 integration end interrupt		390	1186			$+618_H$	0 to 15(15)	
	ASI2 ch2 integration end interrupt		391	1187			$+61C_H$	0 to 15(15)	
	ASI3 ch3 integration end interrupt		392	1188			$+620_H$	0 to 15(15)	
	ASI4 ch4 integration end interrupt		393	1189			$+624_H$	0 to 15(15)	
	ASI5 ch5 integration end interrupt		394	118A			$+628_H$	0 to 15(15)	
	ASI6 ch6 integration end interrupt		395	118B			$+62C_H$	0 to 15(15)	
	ASI7 ch7 integration end interrupt		396	118C			$+630_H$	0 to 15(15)	
	ASI8 ch8 integration end interrupt		397	118D			$+634_H$	0 to 15(15)	
	ASI9 ch9 integration end interrupt		398	118E			$+638_H$	0 to 15(15)	
	ASI10 ch10 integration end interrupt		399	118F			$+63C_H$	0 to 15(15)	
DS-ADC	DSADEn DSADCnAD error interrupt (n = 0 to 7)		400	1190			$+640_H$	0 to 15(15)	
FlexRay	FlexRay 0 interrupt	√	401	1191			$+644_H$	0 to 15(15)	
	FlexRay 1 interrupt	√	402	1192			$+648_H$	0 to 15(15)	
	Timer 0 interrupt	√	403	1193			$+64C_H$	0 to 15(15)	
	Timer 1 interrupt	√	404	1194			$+650_H$	0 to 15(15)	
	Timer 2 interrupt	√	405	1195			$+654_H$	0 to 15(15)	
	FIFO transfer interrupt	√	406	1196			$+658_H$	0 to 15(15)	
	FIFO transfer warning interrupt	√	407	1197			$+65C_H$	0 to 15(15)	
	Output transfer warning interrupt	√	408	1198			$+660_H$	0 to 15(15)	
	Output transfer interrupt	√	409	1199			$+664_H$	0 to 15(15)	
	Input queue full interrupt	√	410	119A			$+668_H$	0 to 15(15)	
	Input queue empty interrupt	√	411	119B			$+66C_H$	0 to 15(15)	
RS-CAN	Reception FIFO interrupt	√	412	119C			$+670_H$	0 to 15(15)	
	Global error interrupt	√	413	119D			$+674_H$	0 to 15(15)	
RS-CAN-Ch0	Transmit/receive FIFO reception completion interrupt	√	414	119E			$+678_H$	0 to 15(15)	
	Error interrupt	√	415	119F			$+67C_H$	0 to 15(15)	
	CAN ch0 transmission interrupt	√	416	11A0			$+680_H$	0 to 15(15)	
RS-CAN-Ch1	Transmit/receive FIFO reception completion interrupt	√	417	11A1			$+684_H$	0 to 15(15)	
	Error interrupt	√	418	11A2			$+688_H$	0 to 15(15)	
	Transmission interrupt	√	419	11A3			$+68C_H$	0 to 15(15)	
RS-CAN-Ch2	Transmit/receive FIFO reception completion interrupt	√	420	11A4			$+690_H$	0 to 15(15)	
	Error interrupt	√	421	11A5			$+694_H$	0 to 15(15)	
	Transmission interrupt	√	422	11A6			$+698_H$	0 to 15(15)	
RS-CAN-Ch3	Transmit/receive FIFO reception completion interrupt	√	423	11A7			$+69C_H$	0 to 15(15)	
	Error interrupt	√	424	11A8			$+6A0_H$	0 to 15(15)	
	Transmission interrupt	√	425	11A9			$+6A4_H$	0 to 15(15)	
	Reserved		426	11AA					
	Reserved		427	11AB					
	Reserved		428	11AC					

Table 6.14 Interrupt Exception Handler and Priority (12/13)

Functional Module	Interrupt Source Name	Level Interrupt Note	Number of Interrupt Channel (EIINT)	Source Code	Offset Address: Direct Branch RINT = 0 Note1	Offset Address: Direct Branch RINT = 1 Note2	Offset Address: Table Reference Note3	Interrupt Priority (Value after reset)	Default Priority
LIN (RLIN2_0)	Transmission end interrupt / reception end interrupt / error detection interrupt	√	429	11AD	The offset address is determined within a range of $+100_H$ to $+1F0_H$ according to the priority, regardless of interrupt channels.	The offset address is always $+100_H$ regardless of priority.	$+6B4_H$	0 to 15(15)	High
	Reserved		430	11AE					
	Reserved		431	11AF					
	Reserved		432	11B0					
	Reserved		433	11B1					
RHSB_0	Data frame transmission start interrupt / data frame transmission execution interrupt	√	434	11B2			$+6C8_H$	0 to 15(15)	
	Transmission start interrupt / command frame transmission execution interrupt	√	435	11B3			$+6CC_H$	0 to 15(15)	
	Emergency frame transmission execution interrupt	√	436	11B4			$+6D0_H$	0 to 15(15)	
	Data reception interrupt	√	437	11B5			$+6D4_H$	0 to 15(15)	
	Upstream error interrupt / time-out detection interrupt / data loss interrupt	√	438	11B6			$+6D8_H$	0 to 15(15)	
RHSB_1	Data frame transmission start interrupt / data frame transmission execution interrupt	√	439	11B7			$+6DC_H$	0 to 15(15)	
	Transmission start interrupt / command frame transmission execution interrupt	√	440	11B8			$+6E0_H$	0 to 15(15)	
	Emergency frame transmission execution interrupt	√	441	11B9			$+6E4_H$	0 to 15(15)	
	Data reception interrupt	√	442	11BA			$+6E8_H$	0 to 15(15)	
	Upstream error interrupt / time-out detection interrupt / data loss interrupt	√	443	11BB			$+6EC_H$	0 to 15(15)	
SCI_0	ERI (reception error)	√	444	11BC			$+6F0_H$	0 to 15(15)	
	RXI (receive data full)		445	11BD			$+6F4_H$	0 to 15(15)	
	TXI (transmit data empty)		446	11BE			$+6F8_H$	0 to 15(15)	
	TEI (transmission end)	√	447	11BF			$+6FC_H$	0 to 15(15)	
SCI_1	ERI (reception error)	√	448	11C0			$+700_H$	0 to 15(15)	
	RXI (receive data full)		449	11C1			$+704_H$	0 to 15(15)	
	TXI (transmit data empty)		450	11C2			$+708_H$	0 to 15(15)	
	TEI (transmission end)	√	451	11C3			$+70C_H$	0 to 15(15)	
SCI_2	ERI (reception error)	√	452	11C4			$+710_H$	0 to 15(15)	
	RXI (receive data full)		453	11C5			$+714_H$	0 to 15(15)	
	TXI (transmit data empty)		454	11C6			$+718_H$	0 to 15(15)	
	TEI (transmission end)	√	455	11C7			$+71C_H$	0 to 15(15)	
SCI_3	ERI (reception error)	√	456	11C8			$+720_H$	0 to 15(15)	
	RXI (receive data full)		457	11C9			$+724_H$	0 to 15(15)	
	TXI (transmit data empty)		458	11CA			$+728_H$	0 to 15(15)	
	TEI (transmission end)	√	459	11CB			$+72C_H$	0 to 15(15)	
CSIH_0	Communication status interrupt (INT_CSIHTIC)		460	11CC			$+730_H$	0 to 15(15)	
	Reception status interrupt (INT_CSIHTIR)		461	11CD			$+734_H$	0 to 15(15)	
	Communication error interrupt (INT_CSIHTIRE)		462	11CE			$+738_H$	0 to 15(15)	
	Job end interrupt (INT_CSIHTIJC)		463	11CF			$+73C_H$	0 to 15(15)	
CSIH_1	Communication status interrupt (INT_CSIHTIC)		464	11D0			$+740_H$	0 to 15(15)	
	Reception status interrupt (INT_CSIHTIR)		465	11D1			$+744_H$	0 to 15(15)	
	Communication error interrupt (INT_CSIHTIRE)		466	11D2			$+748_H$	0 to 15(15)	
	Job end interrupt (INT_CSIHTIJC)		467	11D3			$+74C_H$	0 to 15(15)	
CSIH_2	Communication status interrupt (INT_CSIHTIC)		468	11D4			$+750_H$	0 to 15(15)	
	Reception status interrupt (INT_CSIHTIR)		469	11D5			$+754_H$	0 to 15(15)	
	Communication error interrupt (INT_CSIHTIRE)		470	11D6			$+758_H$	0 to 15(15)	
	Job end interrupt (INT_CSIHTIJC)		471	11D7			$+75C_H$	0 to 15(15)	

Table 6.14 Interrupt Exception Handler and Priority (13/13)

Functional Module	Interrupt Source Name	Level Interrupt Note	Number of Interrupt Channel (EIINT)	Source Code	Offset Address: Direct Branch RINT = 0 Note1	Offset Address: Direct Branch RINT = 1 Note2	Table Reference Note3	Interrupt Priority (Value after reset)	Default Priority
CSIH_3	Communication status interrupt (INT_CSIHTIC)		472	11D8	The offset address is determined within a range of $+100_H$ to $+1F0_H$ according to the priority, regardless of interrupt channels.	The offset address is always $+100_H$ regardless of priority.	$+760_H$	0 to 15(15)	High
	Reception status interrupt (INT_CSIHTIR)		473	11D9			$+764_H$	0 to 15(15)	
	Communication error interrupt (INT_CSIHTIRE)		474	11DA			$+768_H$	0 to 15(15)	
	Job end interrupt (INT_CSIHTIJC)		475	11DB			$+76C_H$	0 to 15(15)	
	Reserved		476	11DC					
	Reserved		477	11DD					
	Reserved		478	11DE					
	Reserved		479	11DF					
DTS	Ch31-0 transfer end interrupt	√	480	11E0			$+780_H$	0 to 15(15)	
	Ch63-32 transfer end interrupt	√	481	11E1			$+784_H$	0 to 15(15)	
	Ch95-64 transfer end interrupt	√	482	11E2			$+788_H$	0 to 15(15)	
	Ch127-96 transfer end interrupt	√	483	11E3			$+78C_H$	0 to 15(15)	
	Ch31-0 transfer count match interrupt	√	484	11E4			$+790_H$	0 to 15(15)	
	Ch63-32 transfer count match interrupt	√	485	11E5			$+794_H$	0 to 15(15)	
	Ch95-64 transfer count match interrupt	√	486	11E6			$+798_H$	0 to 15(15)	
	Ch127-96 transfer count match interrupt	√	487	11E7			$+79C_H$	0 to 15(15)	
	Reserved		488	11E8					
DFE	Error interrupt		489	11E9			$+7A4_H$	0 to 15(15)	
	Reserved		490	11EA					
	Reserved		491	11EB					
	Reserved		492	11EC					
	Reserved		493	11ED					
	Reserved		494	11EE					
	Reserved		495	11EF					
	Reserved		496	11F0					
	Reserved		497	11F1					
	Reserved		498	11F2					
	Reserved		499	11F3					
	Reserved		500	11F4					
	Reserved		501	11F5					
	Reserved		502	11F6					
	Reserved		503	11F7					
	Reserved		504	11F8					
	Reserved		505	11F9					
	Reserved		506	11FA					
	Reserved		507	11FB					
	Reserved		508	11FC					
	Reserved		509	11FD					
	Reserved		510	11FE					
	Reserved		511	11FF					

Note: To clear an interrupt request of level interrupt, the status register in each module should be cleared during interrupt processing by software . To clear the status register, see **Section 3.5.1, Synchronization of Store Instruction Completion and Subsequent Instruction Generation**. The EICTn bit in EICn is set to 1. The EIRFn bit in EICn cannot be cleared by software.

Notes 1, 2, and 3: See **Section 6.4, Interrupt Exception Handler and Priority Operations**.

Note 1. See ***RH850/E1x Flash Memory User's Manual Hardware Interface***.

6.5 Operation

6.5.1 External Interrupts (NMI/IRQ)

NMI and IRQ interrupts are input from external devices. Four detection methods for these interrupts are selectable (2 types for NMI and 4 types for IRQ). For the interrupt detection flow, see **Section 6.5.6, Interrupt Processing Flow**.

6.5.2 Inter-Processor Interrupt

For the operation of software interrupts, see **Section 3.2.3.1, Inter-Processor Interrupt Control Registers** and **Section 6.5.6, Interrupt Processing Flow**.

6.5.3 Software Interrupt

For the operation of software interrupts, see **Section 6.2.9, SINTR0 to SINTR3 — Software Interrupt Registers** and **Section 6.5.6, Interrupt Processing Flow**.

6.5.4 DTS Interrupt Merge Function

Up to 128 transfer end interrupts and up to 128 transfer count match interrupts are aggregated into one interrupt in units of 32 interrupts.

When multiple interrupt source conditions are satisfied, only the single bit on the lower-order side of the corresponding bits in the status registers (PINT0 to PINT7) is set to judge which interrupt is accepted. For the flow of interrupts in the merging of DTS interrupts, see **Section 6.5.6, Interrupt Processing Flow**.

Table 6.15 DTS Interrupt-Related Registers

Interrupt Source	CH	Status Register	Clear Register
DTS transfer end interrupt	0 to 31	PINT0	PINTCLR0
	32 to 63	PINT1	PINTCLR1
	64 to 95	PINT2	PINTCLR2
	96 to 127	PINT3	PINTCLR3
DTS transfer count match interrupt	0 to 31	PINT4	PINTCLR4
	32 to 63	PINT5	PINTCLR5
	64 to 95	PINT6	PINTCLR6
	96 to 127	PINT7	PINTCLR7

6.5.5 ATU-IV/TSG2 Merge Function

ATU-IV timer interrupts and TSG2 interrupts are shared, and functional restrictions shown in **Table 6.16** are provided for these interrupts.

Table 6.17 shows combinations of share.

Table 6.16 Functional Restrictions for Interrupts when Using ATU-IV Timer D and TSG2

ATU-IV timerD (4, 5, 6)				TSG2*1		
(1)	(2)	(3)	(4)	(5)	(6)	
ch4	ch5	ch6	For the peripheral trigger	TSG2_0	TSG2_1	Functional Restriction
√	√	√	√	×	×	TSG2 is not available when ATU-IV is used.
×	×	×	×	√	√	ATU-IV timer D is not available when TSG2 is used.
×	×	×	√	√	√	ATU-IV (timer D4 to 6) is not available when TSG2 is used, but the peripheral trigger function of ATU-IV is available.
×	×	√	√	√	×	Combination of channels for ATU-IV and TSG2 is possible except for some channels.
√	×	×	√	×	√	

Note 1. Cycle interrupt, peak interrupt, and trough interrupt of TSG2 are not merged and can be independently used.

Table 6.17 Sharing of ATU-IV and TSG2

ATU-IV	TSG2*1	Sharing
(1) and (2)*2	(5)	OR
(2)*3 and (3)	(6)	OR

Note 1. Cycle interrupt, peak interrupt, and trough interrupt of TSG2 are not merged and can be independently used.

Note 2. Compare match interrupt (eight) and Overflow interrupt (two) of Timer D ch5

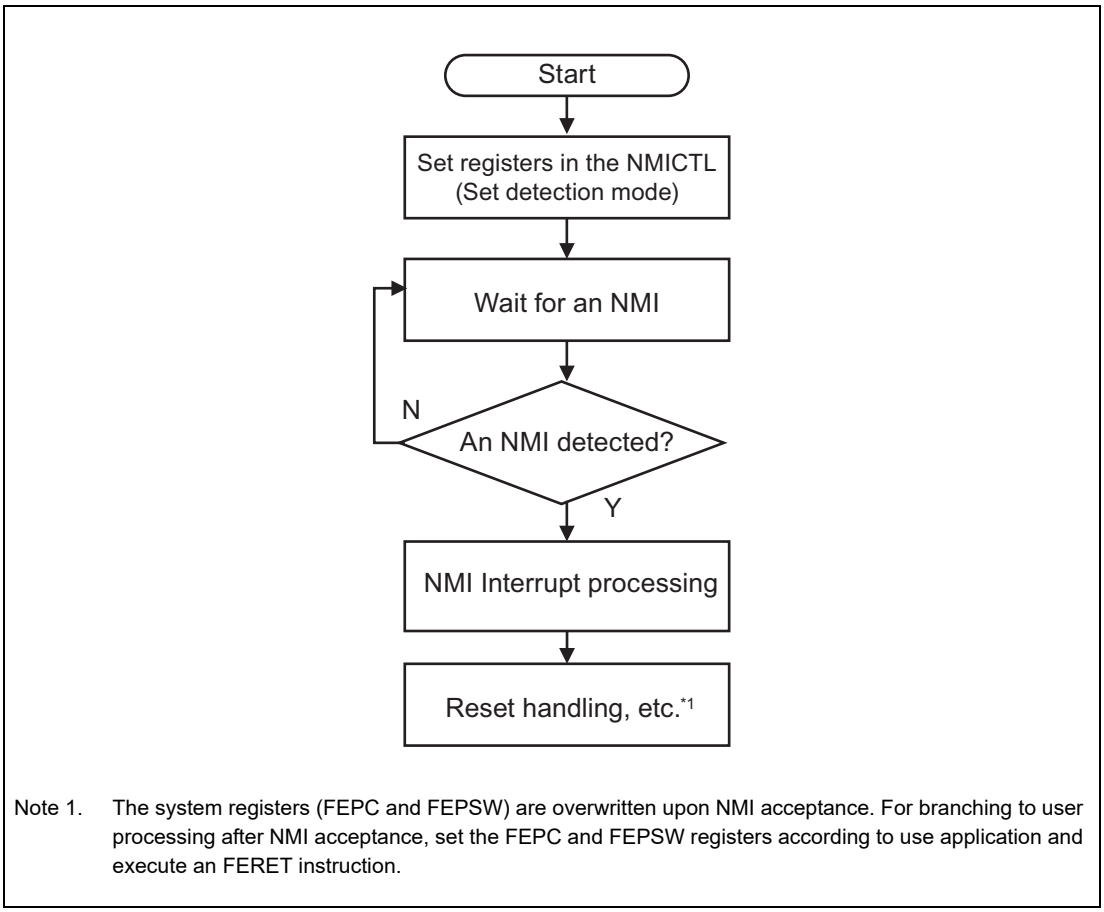
Note 3. Down-counter underflow interrupt of Timer D ch5 (four)

6.5.6 Interrupt Processing Flow

6.5.6.1 NMI Processing Flow

Figure 6.1 shows an example of NMI processing flow.

- Select an NMI detecting method (falling edge or rising edge) by setting the NMICTL register.
- After an NMI has been detected, an interrupt request is sent to the INTC.
- NMI interrupts are accepted at the high priority even when other FE level interrupt has been generated. NMI interrupts cannot be masked regardless of the state of PSW.NP in the CPU system register. NMI interrupts are non-maskable interrupts from which return or recovery is not possible.



Note 1. The system registers (FEPC and FEPSW) are overwritten upon NMI acceptance. For branching to user processing after NMI acceptance, set the FEPC and FEPSW registers according to use application and execute an FERET instruction.

Figure 6.1 Example of NMI Processing Flow

6.5.6.2 External Interrupt Processing Flow

Figure 6.2 shows an example of IRQ (external interrupt) processing flow.

- Select an IRQ detecting method (edge detection or level detection) by setting the EXINTCTL register.
- After detection of an IRQ, an interrupt request is issued to the INTC.
- After recovery on completion of interrupt processing within the INTC in the case of level detection, confirm that the IRQn pin is negated before issuing the instruction to return from the interrupt.
- After recovery on completion of interrupt processing within the INTC in the case of edge detection, clear the interrupt request bit in the EXINTSTR register, and then issue the instruction to return from the interrupt.

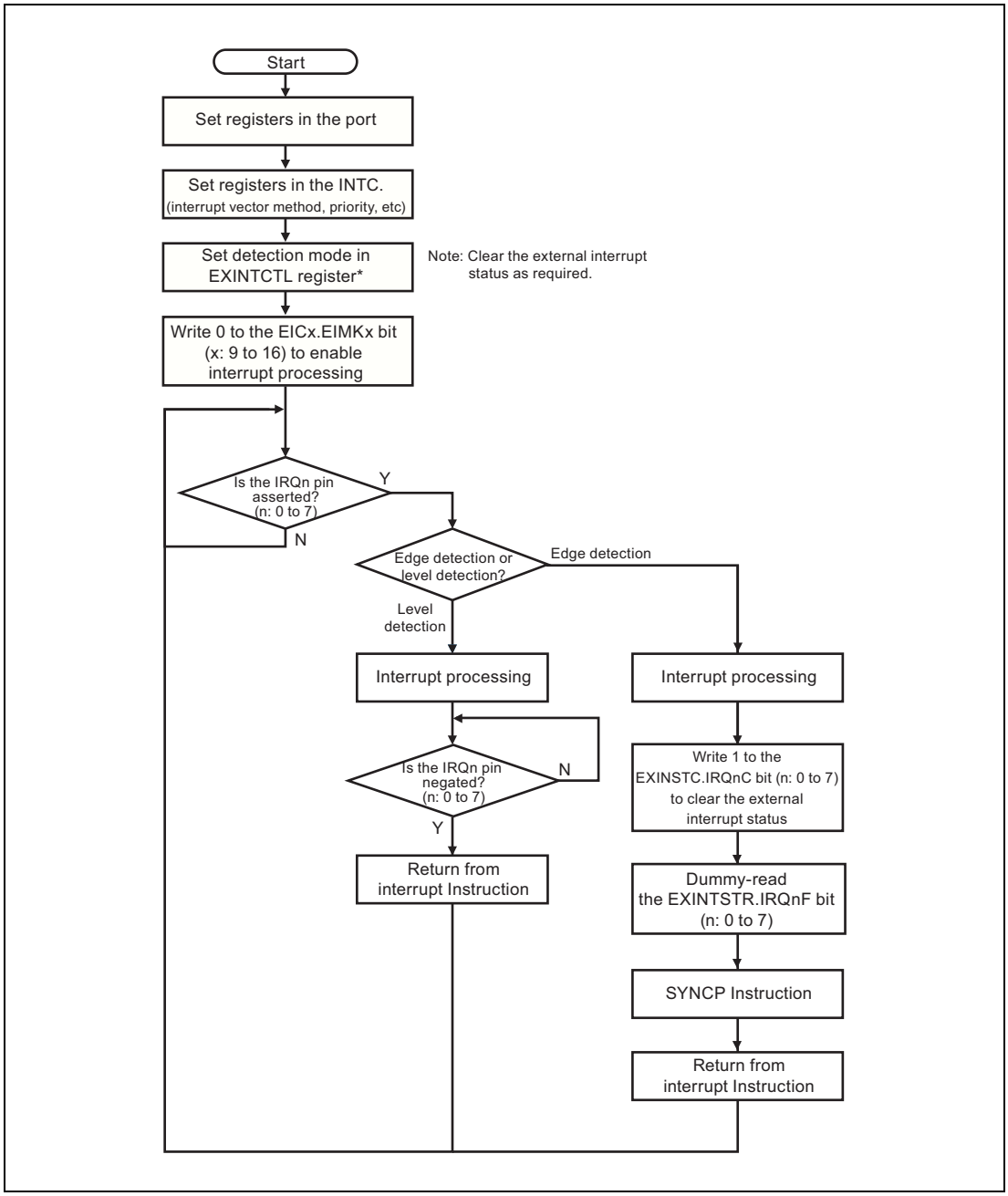


Figure 6.2 Example of External Interrupt Processing Flow

6.5.6.3 Inter-Processor Interrupt Flow

Figure 6.3 shows a flow example of inter-processor interrupt processing.

- Inter-processor interrupting generates an interrupt request by writing 1 to applicable bits of PE to which interrupts of the inter-PE interrupt registers (IPIR0 to IPIR3) are requested.
- The settings of interrupt request of the inter-PE interrupt registers (IPIR0 to IPIR3) are automatically cleared to 0 after notification of an interrupt request is complete.

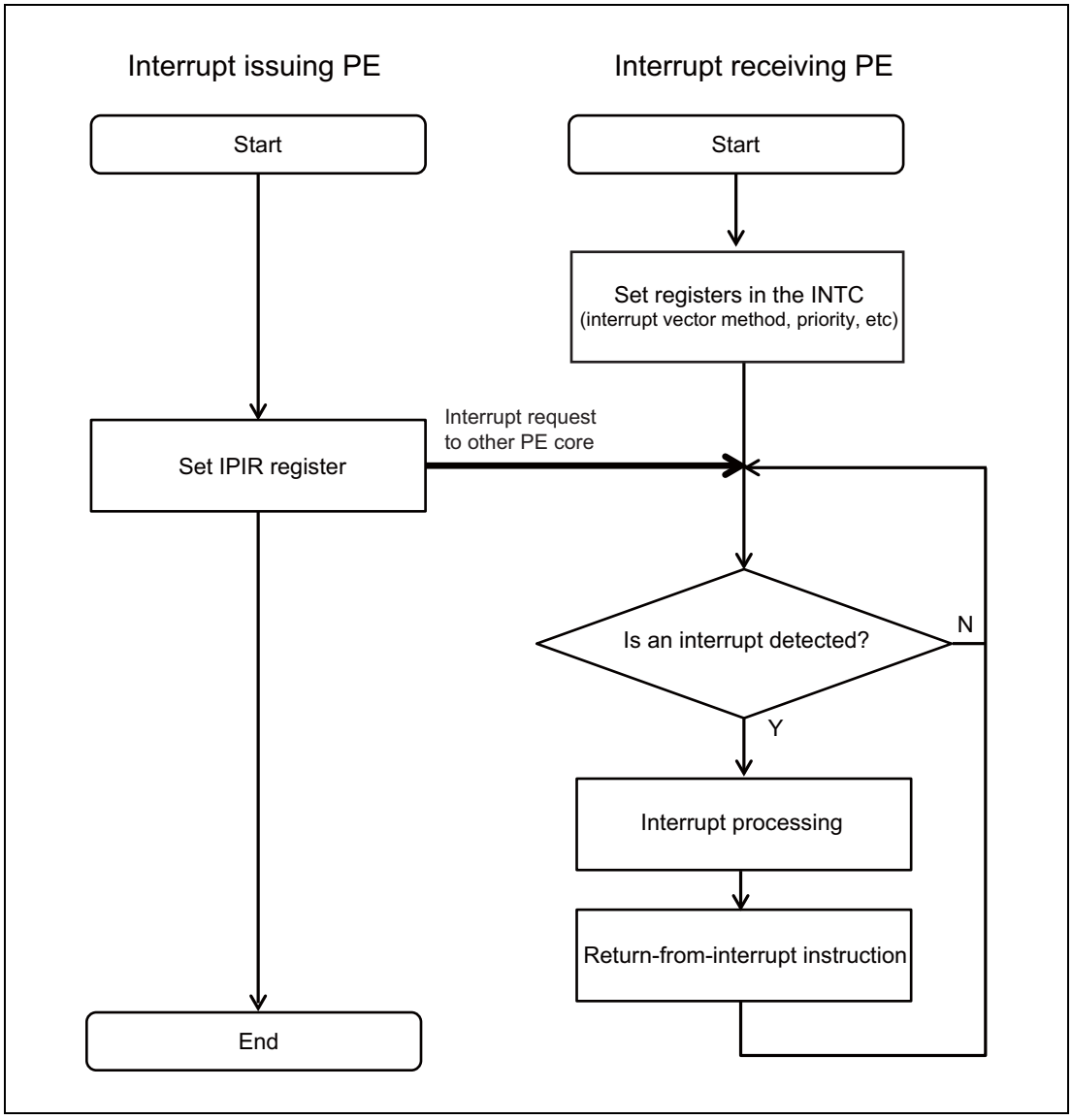


Figure 6.3 Example of Inter-Processor Interrupt Processing Flow

6.5.6.4 Software Interrupt Processing Flow

Figure 6.4 shows the flow example of software interrupt processing.

- Software interrupt requests are controlled by writing 00_H or 01_H to the counter registers (SINTR0 to SINTR3).
- Writing 00_H leads to the counter's value being decremented by 1.
- Writing 01_H leads to the counter's value being incremented by 1.
- If the incremented counter value is 1 or above, an interrupt request for the INTC is generated.
- Decrement the counter by 1 during interrupt processing in the INTC, and if SINTRn is 00_H after issuing the instruction to return from the interrupt, wait for the writing of 01_H to SINTRn.

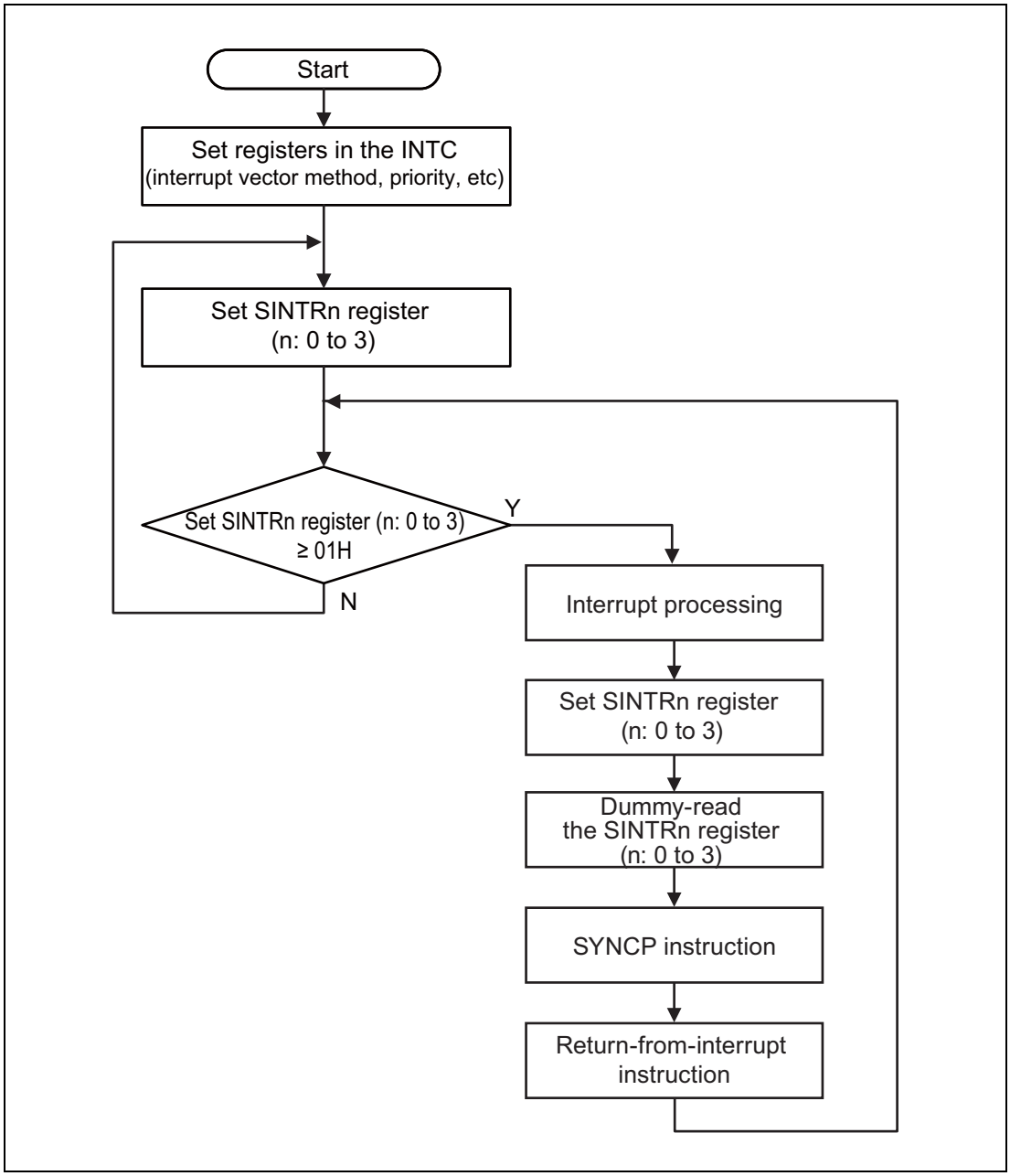


Figure 6.4 Example of Software Interrupt Processing Flow

6.5.6.5 DTS Interrupt Processing Flow

Figure 6.5 shows an example of DTS interrupt processing flow.

- When only one interrupt request is generated out of bundled 32 interrupt sources
  - The bit corresponding to the interrupt request in the PINTn register is set to 1 and an interrupt request is output.
  - On completion of other interrupt processing, write 1 to the interrupt clear register (PINTCLRn) to clear the interrupt request before issuing the return from interrupt instruction, then wait for the next one.
- When multiple interrupt sources are generated out of bundled 32 interrupt sources
  - The bit on the lower-order side is extracted out of bits with interrupt request and only the extracted bit in the PINTn register is set to 1, and an interrupt request is output.
  - On completion of other interrupt processing, write 1 to the target bit in the interrupt clear register (PINTCLRn) to clear the interrupt request, then issue the return from interrupt instruction.
  - After clearing the interrupt request bits for which interrupt processing has proceeded, in the same way as the previous time, the bit on the lower-order side of the PINTn register that corresponds to an interrupt request is extracted and an interrupt request is issued.
  - These steps are repeated until all interrupt sources bundled into 32 bits are cleared.

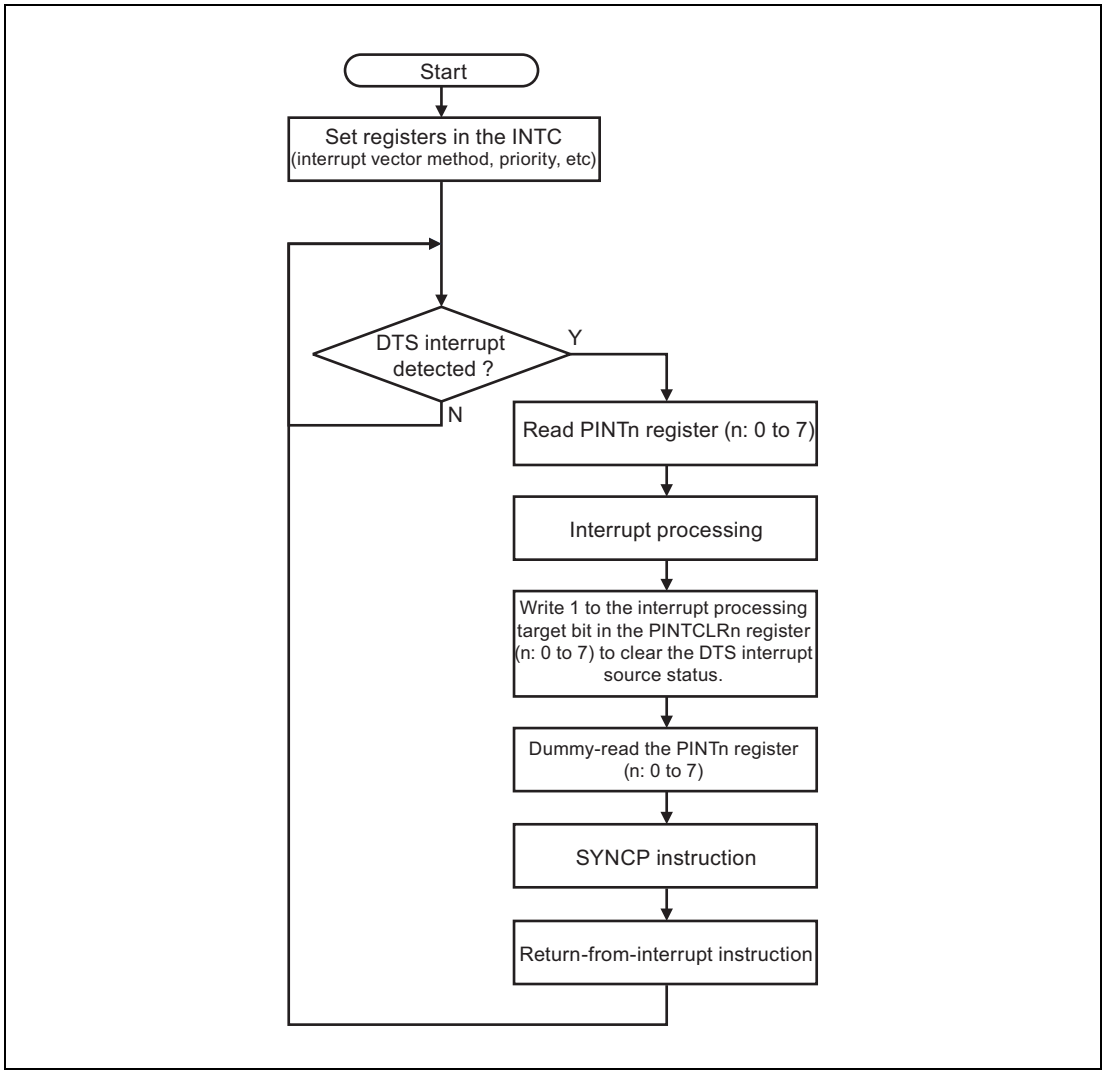


Figure 6.5 Example of DTS Interrupt Processing Flow

6.6 Interrupt Response Times

Table 6.18 Interrupt Response Times (min.)

Interrupt Request Source			Number of Cycles for Processing				
Target	INTC Connection	Operating Clock	Synchro-nization	INTC2	INTC1	In CPU1/ PCU	Total*1
CPU1	Directly input to INTC1	High-speed/low-speed peripheral clock	0	—	2 × Iϕ <1 × Iϕ>	Refer to the descriptions under CPU1 below	7 × Iϕ
		Unmodulated high-speed peripheral clock	—				—
		Unmodulated low-speed peripheral clock	5 × Pϕ <2 × Pϕ>				5 × Pϕ + 7 × Iϕ
	Input via INTC2	High-speed/low-speed peripheral clock	0	3 × Pϕ + 1 × Iϕ <2 × Pϕ + 1 × Iϕ>	—		3 × Pϕ + 6 × Iϕ
		Unmodulated high-speed peripheral clock	4 × Pϕ <2 × Pϕ>				7 × Pϕ + 6 × Iϕ
		Unmodulated low-speed peripheral clock	5 × Pϕ <2 × Pϕ>				8 × Pϕ + 6 × Iϕ
PCU	Directly input to INTC1	High-speed/low-speed peripheral clock	0	—	2 × Pϕ <1 × Pϕ>	Refer to the descriptions under PCU below	9 × Pϕ
		Unmodulated high-speed peripheral clock	—				—
		Unmodulated low-speed peripheral clock	5 × Pϕ <2 × Pϕ>				14 × Pϕ
	Input via INTC2	High-speed/low-speed peripheral clock	0	4 × Pϕ <3 × Pϕ>	—		11 × Pϕ
		Unmodulated high-speed peripheral clock	4 × Pϕ <2 × Pϕ>				15 × Pϕ
		Unmodulated low-speed peripheral clock	5 × Pϕ <2 × Pϕ>				16 × Pϕ

Notes 1. The numbers in < > indicate the numbers of cycles in the case of level detection.

Notes 2. Iϕ = CLK_CPU
Pϕ = 4 Iϕ at 320 MHz,
= 3 Iϕ at 240 MHz

Note 1. In the case of edge detection by fixed vector method a)

Vector method	Cache HIT/MISS	In CPU1 (320 MHz)	In CPU1 (240 MHz)	Vector method	In PCU (320 MHz)	In PCU (240 MHz)
Fixed vector method	a) ISR entry I$ hit	5 × Iϕ	5 × Iϕ	a) Fixed vector method	7 × Pϕ	8 × Pϕ
	b) ISR entry I$ missed	11 × Iϕ	10 × Iϕ			
Vector table reference method	c) Vector code Flash assigned, ISR entry I$ hit	15 × Iϕ	14 × Iϕ	b) Vector table reference method Code flash assigned	14 × Pϕ	15 × Pϕ
	d) Vector code Flash assigned, ISR entry I$ missed	21 × Iϕ	19 × Iϕ			

Note: Iϕ = CLK_CPU
Pϕ = 4 Iϕ at 320 MHz,
= 3 Iϕ at 240 MHz

6.7 Using Interrupt Request Signals to Initiate Data Transfer

An interrupt request signal activates the DMAC/DTS to perform data transfer.

For details, see Section 7, DMA.

Section 7 DMA

7.1 Overview

7.1.1 Overview

Direct memory access (DMA) is used to access data without going through the CPU.

DMA consists of two types of DMA transfer modules: DMAC and DTS. A DMAC includes registers for storing transfer information, and a DTS stores transfer information in the dedicated RAM (DTSRAM). DMA has two 8-channel DMAC modules and one 128-channel DTS module.

In this manual, DTFR denotes the function to select among hardware DMA transfer sources for a DMAC and retain the DMA transfer request, and DTSFSL denotes the function to retain a DMA transfer request for each DTS channel. The DTFR can handle 128 types of hardware DMA transfer sources, and the DTSFSL can handle 128 types of them.

The address space that can be used for DMA transfer is a 4 GB address space represented by a 32-bit address. For information about which resource is assigned to a particular area in the 4 GB address space and which area is accessible from DMA, see **Section 4, Address Space**.

7.1.2 Term Definition

Table 7.1 shows the terms used in this section.

Table 7.1 List of Term Definitions

Term	Meaning
DMA transfer	A general term for data transfer carried out by DMA.
DMA cycle	A series of actions that consist of reading an amount of data specified by the transfer size (8/16/32/64/128 bits) from the address specified by the source address and writing it to the address specified by the destination address. The first half of the DMA cycle (reading part) is called a read cycle, and the second half (writing part) is called a write cycle.
Hardware DMA transfer source	A trigger for a DMA transfer request given by an internal peripheral device.
Hardware DMA transfer request	A DMA transfer request generated by a hardware DMA transfer source.
Software DMA transfer request	A DMA transfer request generated by software writing to a register.
DMA transfer request	A trigger to start DMA transfer with a DMAC and DTS.
Transfer information (TI)	The information required for DMA transfer, including the source address, destination address, transfer data size, and transfer count. The transfer information for a DTS is specifically termed as TI.
DTSRAM	RAM used by a DTS to store the transfer information.
Single transfer	DMA transfer consisting of one DMA cycle started by one DMA transfer request.
Block transfer 1	DMA transfer consisting of a number of DMA cycles specified by the transfer count in the transfer information, started by one DMA transfer request.
Block transfer 2	DMA transfer consisting of a number of DMA cycles specified by the address reload count in the transfer information, started by one DMA transfer request.
Block transfer	A general term for both block transfer 1 and block transfer 2.
Last transfer	The DMA cycle carried out when the transfer count in the transfer information is 1.
Address reload transfer	The DMA cycle carried out when the address reload count in the transfer information is 1 if the reload function 2 is used.
Suspension	An action of pausing DMA transfer during block transfer. You can resume DMA transfer after suspension.
Resume	An action of resuming suspended DMA transfer.
Aborting transfer	An action of aborting DMA transfer in the middle. You cannot resume DMA transfer after that.

7.2 DMA Function

7.2.1 Basic Operation of DMA Transfer

7.2.1.1 Transfer Mode

DMA has three transfer modes.

Single Transfer

One DMA cycle is executed when a DMA transfer request is accepted.

Block Transfer 1

A number of DMA cycles specified in the transfer count register are executed when a DMA transfer request is accepted.

Block Transfer 2

A number of DMA cycles specified by the address reload count are executed when a DMA transfer request is accepted. If the address reload count is larger than the value in the transfer count register, a number of DMA cycles specified in the transfer count register are executed.

7.2.1.2 Executing a DMA Cycle

In a DMA cycle, dual-address transfer (two-cycle transfer) proceeds.

The DMA controller always executes a write cycle after a read cycle is complete. For example, if the transfer data size is 128 bits, a write cycle is executed after a read cycle for the 128-bit data is finished. A write cycle never starts in the middle of a read cycle.

7.2.1.3 Updating Transfer Information

When a DMA cycle is executed, DMA updates transfer information as follows.

Source Address and Destination Address

Transfer information will be updated as described in **Table 7.2** according to the settings for the source address and destination address and the settings in the transfer control register such as the count directions of source address and destination address and transfer data size.

Table 7.2 Updating the Source Address and the Destination Address

Direction of Counting	Transfer Data Size	Address after Update
Increment	8 bit	(address before update) + 0000 0001_H
	16 bit	(address before update) + 0000 0002_H
	32 bit	(address before update) + 0000 0004_H
	64 bit	(address before update) + 0000 0008_H
	128 bit	(address before update) + 0000 0010_H

Table 7.2 Updating the Source Address and the Destination Address

Direction of Counting	Transfer Data Size	Address after Update
Decrement	8 bit	(address before update) – 0000 0001_H
	16 bit	(address before update) – 0000 0002_H
	32 bit	(address before update) – 0000 0004_H
	64 bit	(address before update) – 0000 0008_H
	128 bit	(address before update) – 0000 0010_H
Fixed	—	Same as the address before update.

When you use the reload function, a special update rule is applied other than the one described in **Table 7.2** for the last transfer and the address reload transfer. For details, see **Section 7.2.3, Reload Function**.

Transfer Count/Address Reload Count

The transfer count is decremented by one for every DMA cycle.

The address reload count is decremented by one for every DMA cycle when the reload function 2 or block transfer 2 is used. When the reload function 2 or block transfer 2 is not used, it is not updated.

When you use the reload function, a special update rule is applied for the last transfer and the address reload transfer. For details, see **Section 7.2.3, Reload Function**.

Other transfer information

Other transfer information is not updated during execution of a DMA cycle.

7.2.1.4 Last Transfer and Address Reload Transfer

The last transfer means a DMA cycle executed when the value in the transfer count register, which shows the remaining number of transfers, is one. The last transfer differs in operation compared to other DMA cycles as follows.

- The transfer completion flag (DCSTn.TC) is set when the last transfer is complete. (Only applicable for a DMAC)
- The channel operation enable (DCENn.DTE) bit is cleared when the last transfer is complete. (Only applicable for a DMAC. When the continuous transfer is disabled.)
- When the transfer completion interrupt output enable is set, a transfer completion interrupt is output when the last transfer is complete.
- When the reload function 1 is enabled, the reload function 1 is executed at the timing of the last transfer. For details, see **Section 7.2.3, Reload Function**.

The address reload transfer means a DMA cycle executed when the reload function 2 is enabled and the address reload count is one. The reload function 2 is executed during the address reload transfer. For details, see **Section 7.2.3, Reload Function**.

7.2.1.5 Transfer Completion Interrupt and Transfer Count Match Interrupt Outputs

DMA can output a transfer completion interrupt and a transfer count match interrupt.

Transfer Completion Interrupt Output

When the transfer completion interrupt output enable (DTCTn.TCE) is set in the transfer control register, a DMAC requests a transfer completion interrupt when the last transfer is complete.

When the transfer completion interrupt output enable (DTTCTn.TCE) is set in the transfer control register, a DTS requests a DTS transfer completion interrupt when the last transfer is complete.

Transfer Count Match Interrupt Output

When the transfer count match interrupt enable (DTCTn.CCE) is set in the transfer control register, a DMAC requests a DMAC transfer count match interrupt when the DMA cycle in which the transfer count compare register and the transfer count have the same value is complete.

When the transfer count match interrupt enable (DTTCTn.CCE) is set in the transfer control register, a DTS requests a DTS transfer count match interrupt at the completion of the DMA cycle in which the transfer count compare register and the transfer count have the same value.

Figure 7.1 shows the operation of the transfer completion interrupt and the transfer count match interrupt.

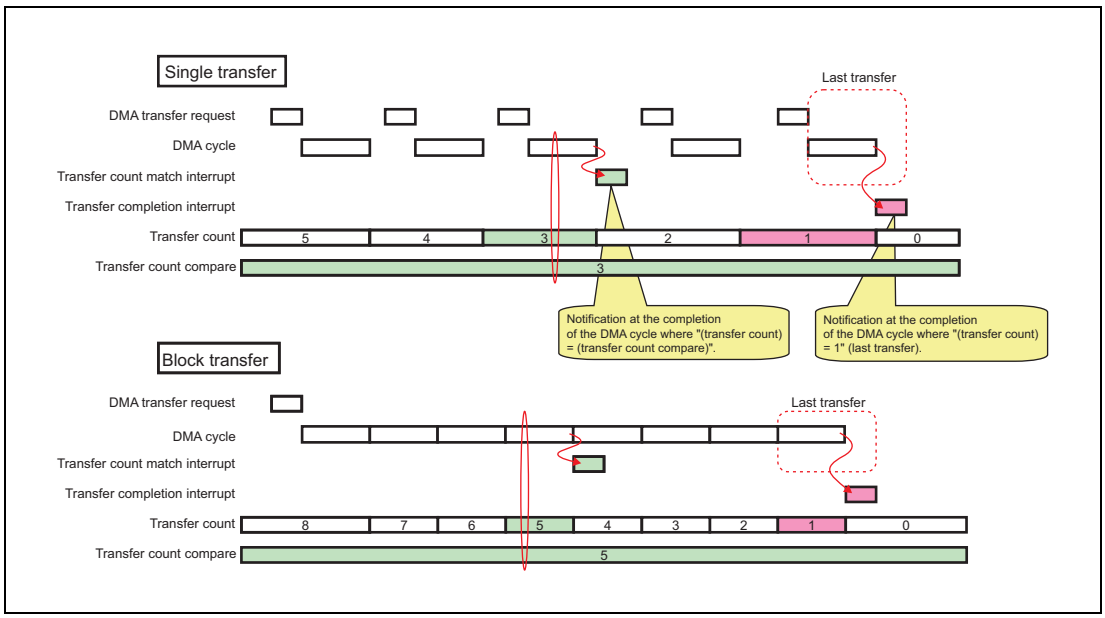


Figure 7.1 Transfer Completion Interrupt and Transfer Count Match Interrupt

7.2.1.6 Continuous Transfer

If the continuous transfer is not used, a DMAC sets the transfer completion flag (DCSTn.TC) and clears the channel operation enable (DCENn.DTE) bit when the last transfer is complete. In this case, a DMA transfer request is not accepted when the request is generated after the completion of the last transfer.

If the continuous transfer is used, the channel operation enable (DCENn.DTE) bit is not cleared when the last transfer is complete, and a DMA transfer request can be accepted even when the transfer completion flag is set. If DMA is used for a case where a specified number of DMA transfers are executed repetitively, software overhead associated with clearing the transfer completion flag and

setting the channel operation enable bit after the completion of the last transfer can be reduced by using the continuous transfer.

The continuous transfer is enabled by setting the continuous transfer enable (DTCTn.MLE) in the DMAC transfer control register.

The continuous transfer is designed to work with the reload function 1. The continuous transfer function itself does not update the source address register, destination address register, and transfer count register. If, after the last transfer is complete, you want to restore the source address register, destination address register, and transfer count register to the state before the DMA transfer starts, use the reload function 1 and set the values of the source address register, destination address register, and the transfer count register before the DMA transfer starts to the reload source address register, reload destination address register, and reload transfer count register respectively.

A DTS does not have a setting corresponding to the continuous transfer enable (DTCTn.MLE) for a DMAC. This is because a DTS does not have bits like the transfer completion flag (DCSTn.TC) and the channel operation enable (DCENn.DTE) a DMAC has.

A DTS does not start DMA transfer when a DMA transfer request is generated while the transfer count is 0. (This corresponds to the case for a DMAC where the continuous transfer is not used.)

If the reload function 1 is used for a DTS and the value other than 0 is reloaded to the transfer count when the last transfer is complete, DMA transfer can start when the next DMA transfer request is accepted. (This corresponds to the case for a DMAC where the continuous transfer is used.)

Figure 7.2 shows an operation of continuous transfer by a DMAC.

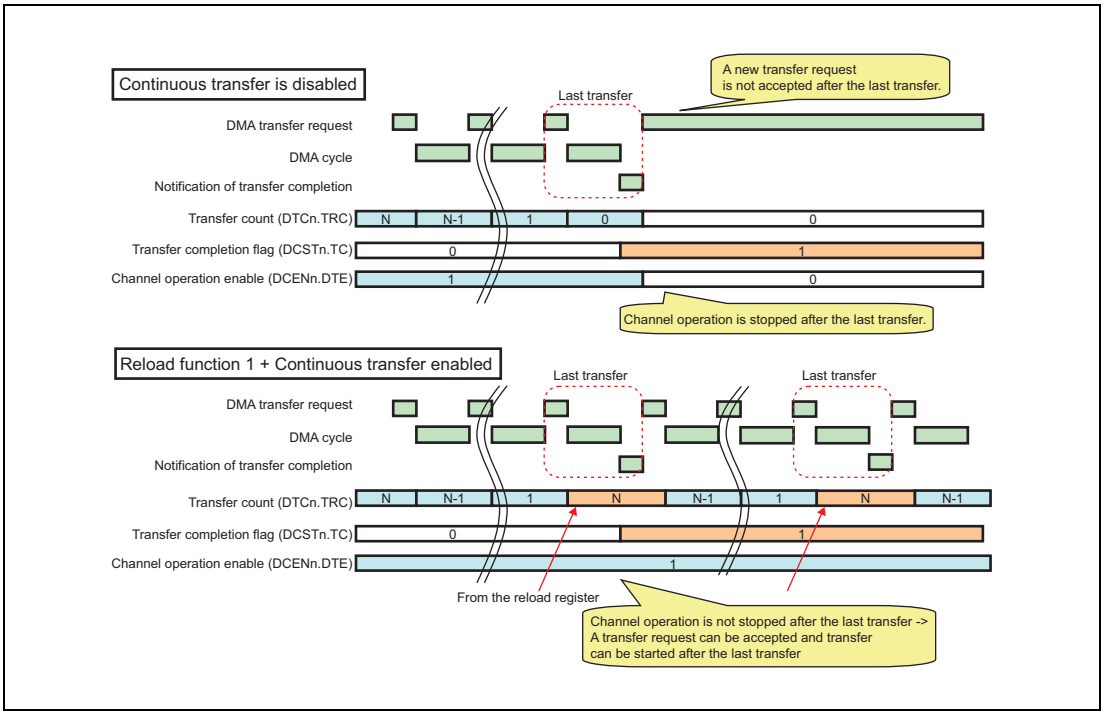


Figure 7.2 Operation of Continuous Transfer by a DMAC

7.2.2 Channel Priority Order

This subsection explains arbitration between multiple DMA channels.

7.2.2.1 DMAC Channel Arbitration

A DMAC selects one channel out of eight channels with arbitration. Arbitration is done according to the fixed priority order. The priority order is “channel 0 > channel 1 > channel 2 > channel 3 > channel 4 > channel 5 > channel 6 > channel 7” for DMAC0, and “channel 8 > channel 9 > channel 10 > channel 11 > channel 12 > channel 13 > channel 14 > channel 15” for DMAC1.

Arbitration is done for every DMA cycle. No arbitration occurs between the read and write of a DMA cycle.

If, at the timing when one DMA cycle in the middle of a block transfer by a channel is complete, there is a DMA transfer request from a channel with a higher priority than the channel, a DMA cycle of the channel with the higher priority will be executed next as the result of arbitration.

If a DMAC executes the block transfer 1 or block transfer 2, DMAC channel arbitration is done for every DMA cycle, and possibly a DMA cycle of another DMAC channel with a higher priority can cut in.

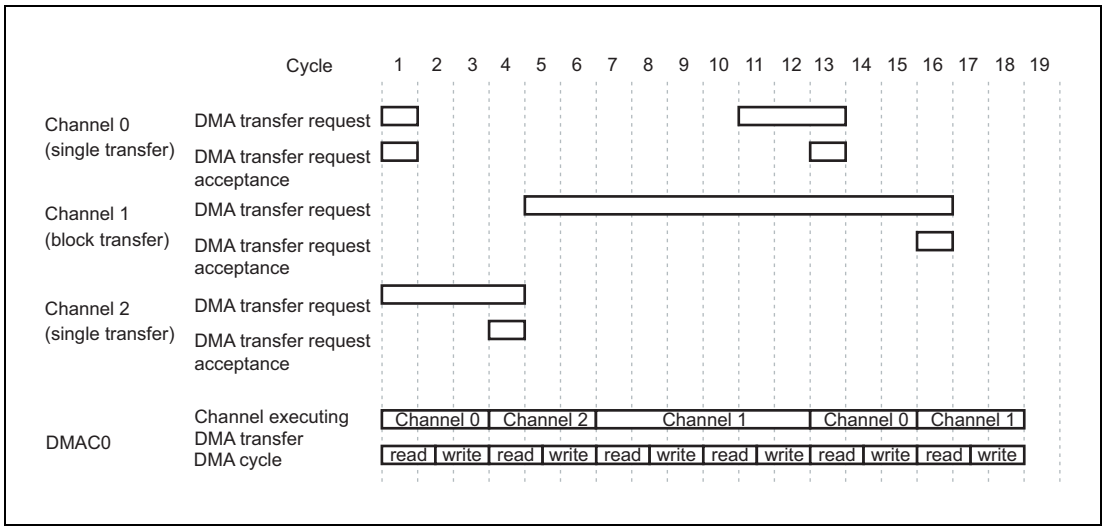


Figure 7.3 DMAC Channel Arbitration

Cycle numbers shown in **Figure 7.3** are for explanation purpose only. They do not indicate an actual number of cycles necessary for executing DMA transfer.

In **Figure 7.3**, DMA transfer requests for channels 0 and 2 are generated at Cycle 1. As a result of arbitration, a DMA cycle for channel 0 starts because its priority is higher. At Cycle 4, a DMA cycle for channel 2 starts. At Cycle 5, a DMA transfer request for channel 1 is generated. The DMA cycle for channel 2 is still ongoing and no arbitration is done at this point. At Cycle 7, a DMA cycle for channel 1 starts. Channel 1 uses block transfer. Another DMA cycle continues at Cycle 10 because there are no DMA transfer requests from other channels. At Cycle 11, a DMA transfer request for channel 0 is generated. The DMA cycle for channel 1 is still ongoing and no arbitration is done at this point. At Cycle 12, the DMA cycle for channel 1 is complete. At Cycle 13, a DMA cycle for channel 0 starts as a result of arbitration between DMA channels 0 and 1.

It should be noted that, even though a block transfer by channel 1 has been already started, a DMA cycle of not channel 1 but channel 0 is executed at Cycle 13 because the priority of the latter is higher.

At Cycle 15, the DMA cycle for channel 0 is complete. At Cycle 16, a DMA cycle for channel 1 starts again. At Cycle 18, the last DMA cycle of the block transfer by channel 1 is complete.

7.2.2.2 DTS Channel Arbitration

If there are DMA transfer requests from multiple DTS channels, the DTSFSL arbitrates those DTS channels. For each DTS channel, a priority can be selected from four levels using DTS channel priority setting registers.

If there are DMA transfer request from multiple DTS channels, the arbitration is done as follows.

1. A channel with a higher priority level in the setting of DTS channel priority setting registers has a priority.
2. If two channels have the same priority level in the setting of DTS channel priority setting registers, a channel with a lower channel number has a priority.

The DTSFSL sends the DTS a DMA transfer request for the channel selected by arbitration. The DTS executes DMA transfer when it accepts the DMA transfer request.

Unlike DMA transfer with a DMAC, DMA transfer with a DTS does not allow arbitration between DTS channels in the middle of a block transfer. That means, even if a DMA transfer request with a higher priority comes during a block transfer for a channel with a lower priority, the DMA transfer with a higher priority does not start until the current block transfer for the channel with a lower priority is complete*1.

Note 1. The timing of completion of the block transfer is when the last transfer for the block transfer 1 or the last transfer or address reload transfer for the block transfer 2 occurs.

When a DTS executes the block transfer 1 or block transfer 2, a DMA cycle of a DTS channel with a higher priority does not take over the ongoing block transfer until the last transfer.

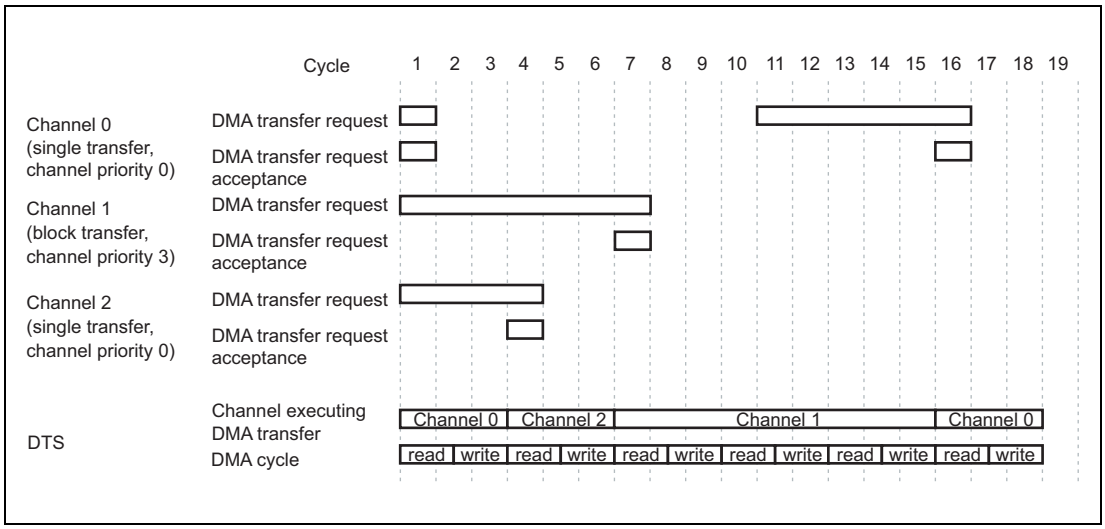


Figure 7.4 DTS Channel Arbitration

Cycle numbers shown in **Figure 7.4** are for explanation purpose only. They do not indicate an actual number of cycles necessary for executing DMA transfer.

In **Figure 7.4**, DMA transfer requests for channels 0, 1, and 2 are generated at Cycle 1. The channel priority for channels 0 and 2 is 0 and is higher than the channel priority for channel 1, which is 3. In addition, if two channels have the same priority, the channel with the smaller channel number has a higher priority. Consequently, the priority order for arbitration is “channel 0 > channel 2 > channel 1”, and a DMA cycle for channel 0 starts because its priority is the highest At Cycle 4, as a result of arbitration between channels 1 and 2, a DMA cycle for channel 2 starts. At Cycle 7, a DMA cycle for channel 1 starts. Channel 1 uses block transfer. Another DMA cycle continues at Cycle 10 because there are no DMA transfer requests from other channels. At Cycle 11, a DMA transfer request for channel 0 is generated. The DMA cycle for channel 1 is still ongoing and no arbitration is done until the block transfer by channel 1 is complete.

At Cycle 15, the block transfer by channel 1 is complete. At Cycle 16, a DMA cycle for channel 0 starts.

7.2.2.3 Interface Arbitration

DMAC0, DMAC1, and DTS work independently and execute DMA transfer.

When DMAC0, DMAC1, and DTS requests are in contention, the interface arbitrates this in round-robin fashion.

7.2.3 Reload Function

7.2.3.1 Overview of the Reload Function

The reload function updates a portion of transfer information, more specifically, the source address, destination address, transfer count, and address reload count, to the predefined values during DMA transfer.

The reload function has two types of functions: reload function 1 and reload function 2.

7.2.3.2 Operation of Reload Function 1

When the reload function 1 is enabled, actions described in **Table 7.3** are executed at the timing of the last transfer according to the reload function 1 setting.

Table 7.3 Operation of Reload Function 1

Reload Function 1 Setting (DTCTn.RLD1M[1:0])	Register	Action at the Last Transfer
00 (Reload function 1 disabled.)	Source address	Not reloaded.
	Destination address	Not reloaded.
	Transfer count	Not reloaded.
	Address reload count	Not reloaded.
01 (Reload function 1 enabled. Reloading source address and transfer count.)	Source address	The reload source address is copied to this.
	Destination address	Not reloaded.
	Transfer count	The reload transfer count is copied to this.
	Address reload count	• If the reload function 2 is disable: Not reloaded. • If the reload function 2 is enabled: The reload address reload count is copied to this.
10 (Reload function 1 enabled. Reloading destination address and transfer count.)	Source address	Not reloaded.
	Destination address	The reload destination address is copied to this.
	Transfer count	The reload transfer count is copied to this.
	Address reload count	• If the reload function 2 is disable: Not reloaded. • If the reload function 2 is enabled: The reload address reload count is copied to this.
11 (Reload function 1 enabled. Reloading source address, destination address, and transfer count.)	Source address	The reload source address is copied to this.
	Destination address	The reload destination address is copied to this.
	Transfer count	The reload transfer count is copied to this.
	Address reload count	• If the reload function 2 is disable: Not reloaded • If the reload function 2 is enabled: The reload address reload count is copied to this.

Figure 7.5 shows an operation of the reload function 1.

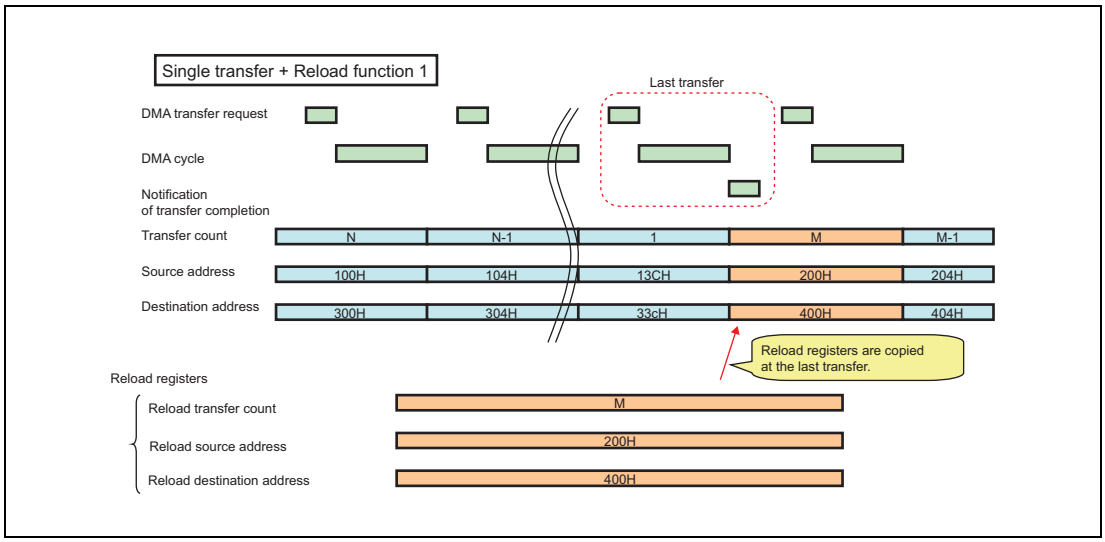


Figure 7.5 Operation of Reload Function 1

7.2.3.3 Reload Function 2

When the reload function 2 is enabled, actions described in **Table 7.4** are executed at the timing of the address reload transfer according to the reload function 2 setting.

Table 7.4 Operation of Reload Function 2

Reload Function 2 Setting (DTCTn.RLD2M[1:0])	Register	Action at the Address Reload Transfer
00 (Reload function 2 disabled.)	Source address	Not reloaded.
	Destination address	Not reloaded.
	Address reload count	Not reloaded.
01 (Reload function 2 enabled. Reloading source address.)	Source address	The reload source address is copied to this.
	Destination address	Not reloaded.
	Address reload count	The reload address reload count is copied to this.
10 (Reload function 2 enabled. Reloading destination address.)	Source address	Not reloaded.
	Destination address	The reload destination address is copied to this.
	Address reload count	The reload address reload count is copied to this.
11 (Reload function 2 enabled. Reloading source address and destination address.)	Source address	The reload source address is copied to this.
	Destination address	The reload destination address is copied to this.
	Address reload count	The reload address reload count is copied to this.

Figure 7.6 shows an operation of the reload function 2.

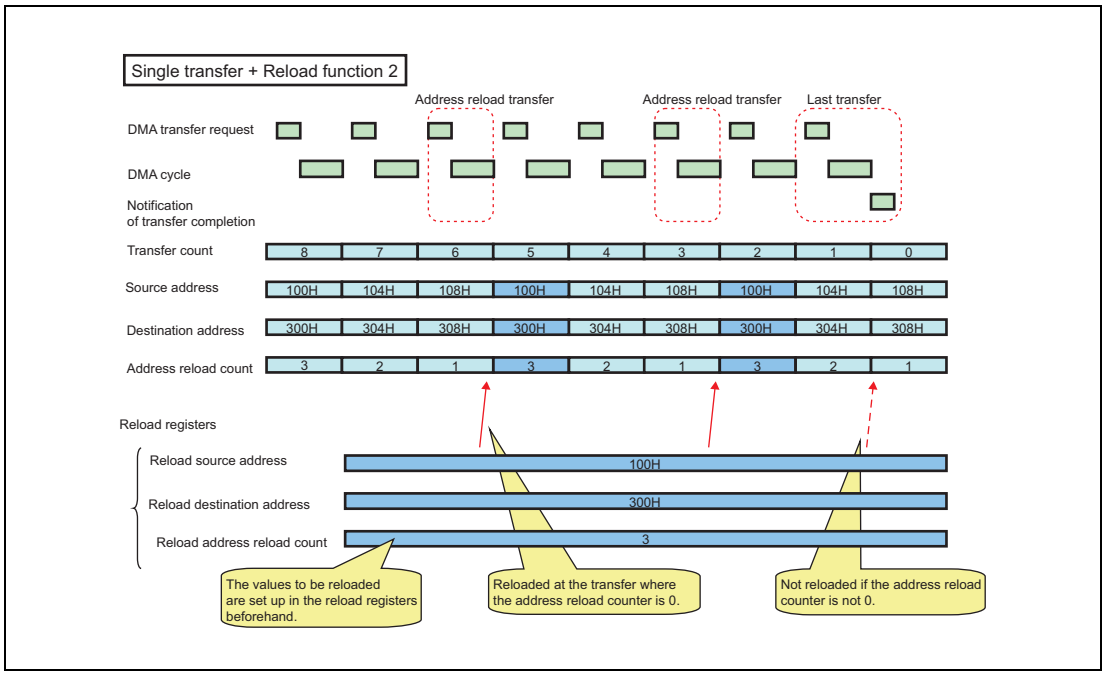


Figure 7.6 Operation of Reload Function 2

Figure 7.7 shows an operation when both the reload function 1 and the reload function 2 are used simultaneously.

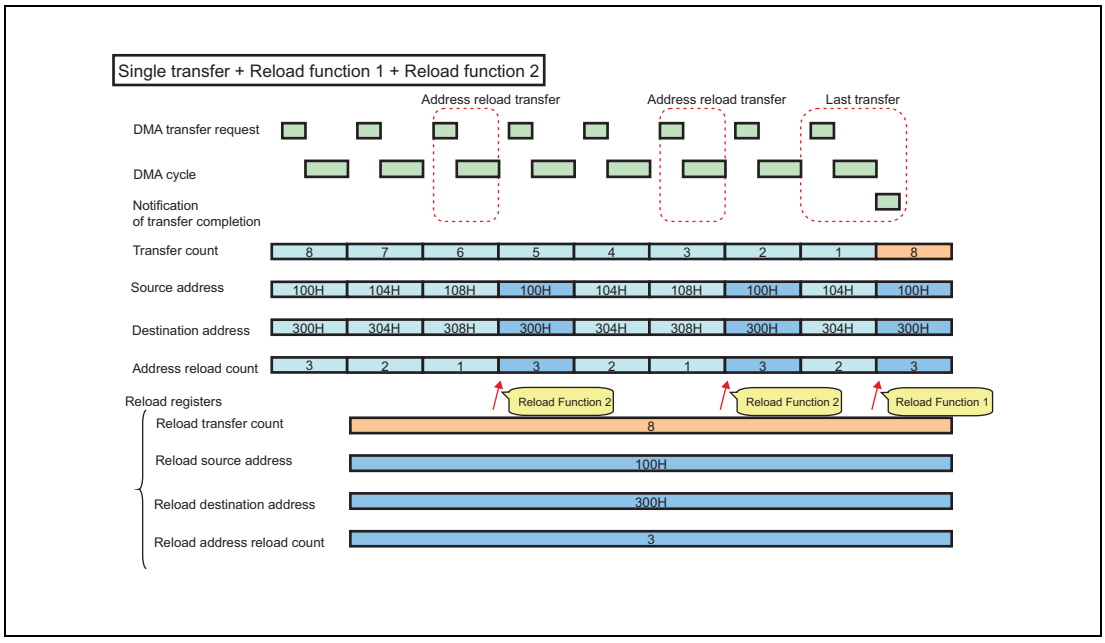


Figure 7.7 Operation when combining the reload function 1 and the reload function 2

7.2.3.4 Timing of Setting DMAC Reload Registers

You can set up the reload source address register, reload destination address register, and reload transfer count register any time (even during DMA transfer). As an exception, if you update the reload source address register, reload destination address register, and reload transfer count register during DMA transfer, there may be a conflict between reloading at the last transfer or address reload transfer and updating the reload register. In order to avoid this conflict, setting up reload registers must be completed before the last transfer or address reload transfer starts.

If you need to update the reload source address register, reload destination address register, and reload transfer count register during DMA transfer, one way to know the right timing of update is to use a DMA transfer count match interrupt. In this case, you must set up the DMA transfer count compare register (DTCCn) so that you can have enough margin for the time necessary to update the reload registers.

7.2.3.5 Timing of Setting DTS Reload Registers

It should be noted that the right timing of setting up the reload source address information, reload destination address information, and reload transfer count information differs depending on the transfer mode.

In the single transfer mode, the TI fetched at the beginning of the last transfer or address reload transfer is used for reload at the completion of the DMA cycle. Therefore, if you use the reload function for single transfer, the reload source address information, reload destination address information, and reload transfer count information in the TI must be set up before the beginning of the last transfer or address reload transfer.

During block transfer, TI is fetched only at the beginning of DMA transfer. The TI fetched at the beginning of the DMA transfer is used for reload at the last transfer or address reload transfer. Therefore, if you use the reload function for block transfer, the reload source address information, reload destination address information, and reload transfer count information in the TI must be set up before the beginning of the DMA transfer. If you update the reload source address information, reload destination address information, and reload transfer count information in the TI in the middle of a block transfer, those new settings will not be used for reload at the completion of the block transfer.

7.2.4 Chain Function

7.2.4.1 Overview

DMA offers a function called chain function. If you use the chain function, the completion of the DMA cycle or last transfer for one channel can trigger a DMA transfer request for another channel. A DMA transfer request for another channel initiated by the chain function is called a chain request.

You can select the condition for generating a chain request from the following two options.

- Always chain: A chain request is generated at the completion of every DMA cycle.
- Chain at the last transfer: A chain request is generated at the completion of the last transfer.

Figure 7.8 shows an operation of the case “always chain”.

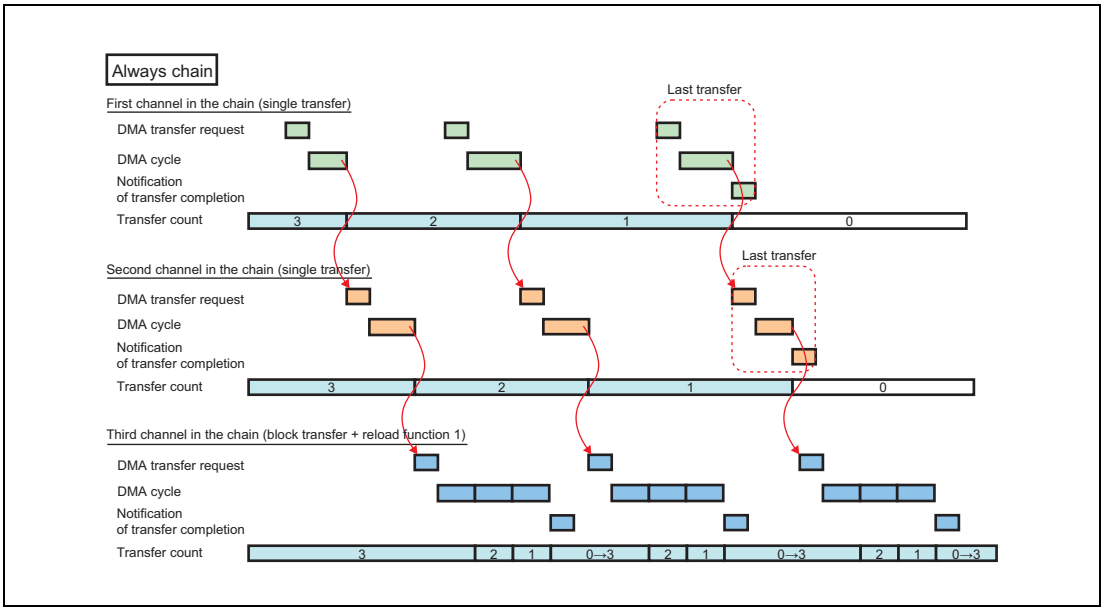


Figure 7.8 Operation of the Case “Always Chain”

Figure 7.9 shows an operation of the case “chain at the last transfer”.

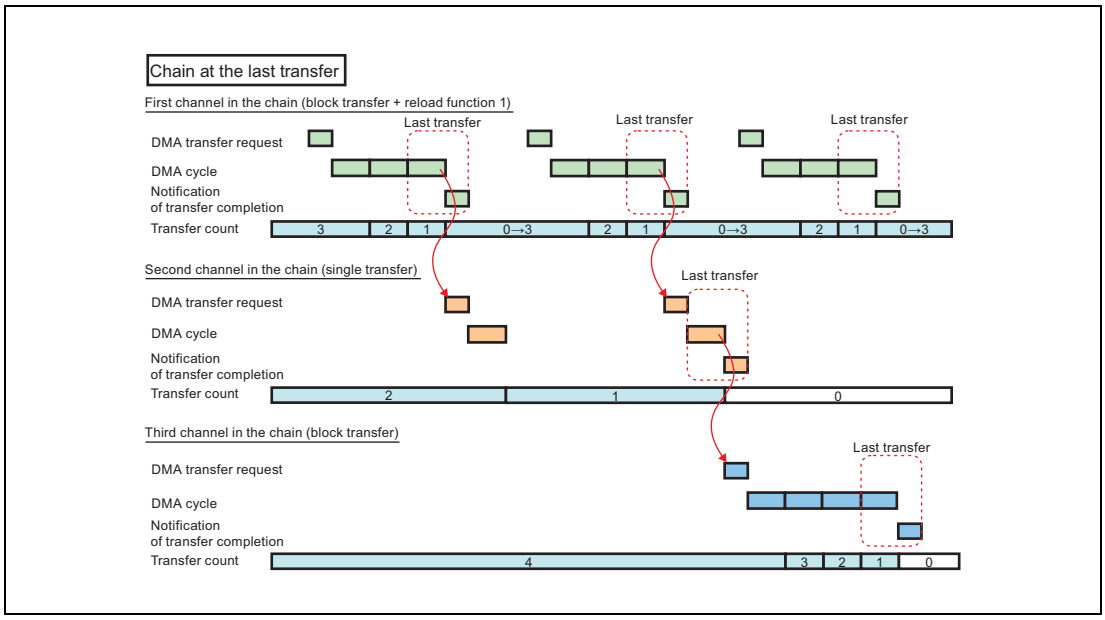


Figure 7.9 Operation of the Case “Chain at the Last Transfer”.

7.2.4.2 Setting Up the Chain Function

For a DMAC, you need to write to the chain enable (DTCTn.CHNE) and the next channel in the chain selection (DTCTn.CHNSEL) in the DMAC transfer control register in order to set up the type of chain function and the next channel number in the chain.

For a DTS, you need to write to the chain enable (DTTCTnnn.CHNE) and the next channel in the chain selection (DTTCTnnn.CHNSEL) in the DTS transfer control register in order to set up the type of chain function and the next channel number in the chain.

7.2.4.3 Caution for Using the Chain Function

The chain function sets the software DMA transfer request flag of the next channel in the chain as a part of its function. Therefore, you need to set up the channel settings of the next channel in the chain in the same way as when the software DMA transfer request is used. If you specify a channel using the hardware DMA transfer request for the next channel in the chain, the chain function does not work.

A channel and its next channel in the chain must belong to the same module (DMAC0, DMAC1, and DTS). You cannot specify a channel in another module for its next channel in the chain.

7.2.5 DMAC Operation

7.2.5.1 Types of DMA Transfer Requests and Assigning DMA Transfer Requests

A DMAC starts DMA transfer by accepting a hardware DMA transfer request or software DMA transfer request. The DMA transfer request selection assignment (DRS) bit in the DMAC transfer control register (DTCTn) determines whether a hardware DMA transfer request or a software DMA transfer request is used.

In the case of a hardware DMA transfer request for a DMAC, one out of 128 hardware DMA transfer sources is selected and assigned for each channel of the DMAC in the DTFR. This assignment is configured in the DTFR setting registers.

7.2.5.2 Generating and Accepting a Software DMA Transfer Request

By setting the software DMA transfer request flag (SR) in the DMAC transfer status register (DCSTn) using the DMAC transfer status set register (DCSTSn), a software DMA transfer request can be generated.

The software DMA transfer request flag is automatically cleared when the DMAC processes the DMA transfer request. The timing when the software DMA transfer request flag is automatically cleared differs depending on the transfer mode of the DMA transfer to be executed.

- In the single transfer mode, the software DMA transfer request flag is cleared whenever the software DMA transfer request is accepted.
- In the block transfer 1 mode, the software DMA transfer request flag is cleared when the last transfer starts.
- In the block transfer 2 mode, the software DMA transfer request flag is cleared when the last transfer or address reload transfer starts.

The software DMA transfer request flag can also be cleared by software using the DMAC transfer status clear register (DTSTCn). When you abort a DMA transfer by a DMAC channel, you must clear the software DMA transfer request flag.

7.2.6 DTS Operation

7.2.6.1 Types of DMA Transfer Requests and Assigning DMA Transfer Requests

A DTS starts DMA transfer by accepting a hardware DMA transfer request or software DMA transfer request. A transfer request for a DTS is retained in the transfer request pending bit of the DTSFSL for each channel.

As for the DTSFSL, both a hardware DMA transfer request and a software DMA transfer request are retained in the same transfer request pending bit. When executing DMA transfer, a DTS does not care whether a DMA transfer request is a hardware DMA transfer request or software DMA transfer request.

In the case of a hardware DMA transfer request for a DTS, each of 128 hardware DMA transfer sources is assigned to one of 128 channels in the DTSFSL in the fixed manner. You cannot change this assignment by, for example, register settings.

7.2.6.2 Generating and Accepting a DMA Transfer Request

When the DTSFSL detects a hardware DMA transfer source input, the DTSFSL sets the transfer request pending bit and retains the hardware DMA transfer source as a DMA transfer request. If the transfer request pending bit is set and the transfer request enable bit (DTFSLnnn.REQEN) in the DTSFSL operation setting register is set, the DTSFSL notifies the DTS of the DMA transfer request.

Software can also generate a DMA transfer request by setting the transfer request pending bit (DTFSTnnn.DRQ) using the DTSFSL transfer request set register (DTFSSnnn).

The DTSFSL can retain only one DMA transfer request per channel. If, while the transfer request pending bit for a channel is set, a new hardware DMA transfer source input for the same channel comes, the new hardware DMA transfer source input is ignored.

When the DTS accepts a DMA transfer request, it notifies of the acceptance of the DMA transfer request.

The transfer request pending bit is automatically cleared when the DTS accepts the DMA transfer request. The DTSFSL clears the transfer request pending bit automatically when the DTS accepts the DMA transfer request regardless of the type of the DMA transfer to be executed by the DTS.

The transfer request pending bit can also be cleared using the DTSFSL transfer request clear register (DTFSCnnn). If the transfer request pending bit of a channel is cleared before the DTS accepts the DMA transfer request, DMA transfer by the channel is not executed.

7.2.6.3 Executing DMA Transfer

When the DTS accepts a DMA transfer request for a channel, the DTS executes DMA transfer on the channel. If there are DMA transfer requests from multiple channels, the DTSFSL arbitrates the DTS channels and picks up one channel for a DMA transfer request.

While the DTS is executing DMA transfer, the DTS transfer status (DTSSTS.DTSACT) bit in the DTS status register is set. In addition, the channel number of the currently ongoing DMA transfer is set in the DTS transfer channel (DTSSTS.DTSACH).

When the DMA transfer is complete or aborted because of DMA transfer error or writing to registers and no channel is currently executing DMA transfer, the DTS transfer status (DTSSTS.DTSACT) bit is cleared.

7.2.6.4 DTSRAM Access

A DTS accesses the DTSRAM when DMA transfer starts and finishes.

A DTS's action of reading transfer information from the DTSRAM when DMA transfer starts is called TI fetch.

A DTS's action of updating the transfer information on the DTSRAM when DMA transfer finishes is called TI write back.

A single transfer performs a TI fetch at the beginning of a DMA cycle and a TI write back at the end of a DMA cycle.

A block transfer performs a TI fetch at the beginning of the first DMA cycle and a TI write back at the end of the DMA cycle that satisfies the block transfer completion condition (the last transfer or address reload transfer).

Therefore, in the case of single transfer, the transfer information on the DTSRAM is updated for each DMA cycle. In the case of block transfer, the transfer information on the DTSRAM is updated after the completion of the block transfer. If software reads the transfer information on the DTSRAM during execution of a block transfer, the transfer information at the beginning of the block transfer is read.

7.3 Suspending, Resuming, and Aborting DMA Transfer and Clearing the DMA Transfer Request

7.3.1 Suspending and Resuming DMA Transfer by Software Control

The DMA control register (DMACTL) is used to suspend DMA transfer for all channels.

When the DMA suspension bit (DMACTL.DMASPD) in the DMA control register is set, DMA puts all channels into the suspended state If all channels are in the suspended state and the DMA suspension bit (DMACTL.DMASPD) in the DMA control register is cleared, DMA restores all channels from the suspended state to the normal state and resumes the DMA transfer on the suspended channel.

When all channels are put into the suspended state, DMA transfer is suspended for all channels without changing the value of the DCENn.DTE bit of each DMAC channel and the DTSCTL1.DTSUST bit of the DTS.

CAUTION

To suspend or resume currently ongoing DMA transfer by the DTS, the DMA transfer request enable bit (DTFSLnnn.REQEN) must be switched to disable and enable arbitration for the given channel. For details, refer to the notes in Section 7.3.3, Suspending, Resuming, and Aborting Transfer by the DTS.

7.3.2 Suspending, Resuming, and Aborting Transfer by a DMAC Channel

You can suspend DMA transfer by a DMAC channel by clearing the channel operation enable bit in the DMAC channel operation enable setting register (DCENn.DTE) or setting the DMA suspension bit in the DMA control register (DMACTL.DMASPD). If a DMA cycle is ongoing, DMA transfer is suspended after the currently ongoing DMA cycle is finished. If you set the DCENn.DTE bit again or clear the DMACTL.DMASPD bit while DMA transfer is suspended, the suspended DMA transfer by the DMAC channel is resumed.

If you want to abort currently ongoing DMA transfer by a DMAC channel, clear the channel operation enable bit in the DMAC channel operation enable setting register (DCENn.DTE), and then clear the hardware DMA transfer request in the DTFR in the case of a hardware DMA transfer request, and clear the software DMA transfer request flag (DCSTn.SR) by using the DMAC transfer request flag clear bit in the DMAC transfer status clear register (DCSTCn.SRC) in the case of a software DMA transfer request.

While the continuous transfer enable bit (DTCTn.MLE) is set, the channel operation enable bit (DCENn.DTE) remains set. Even if the channel operation enable bit (DCENn.DTE) is cleared by software during the DMA cycle of the last transfer, the function of the continuous transfer enable bit (DTCTn.MLE) is given priority and the channel operation enable bit (DCENn.DTE) is set again after the completion of the last transfer.

If you want to suspend the DMAC channel while continuous transfer is in use, clear the continuous transfer enable bit (DTCTn.MLE) of the DMAC transfer control register, and then clear the channel operation enable bit (DCENn.DTE) of the DMAC channel operation enable setting register to suspend the DMA transfer. When and only when this operation is performed, writing to the DMAC transfer control register (DTCTn) is allowed while channel operation is enabled (DCENn.DTE = 1).

Figure 7.10 shows an example of suspending, resuming, and aborting transfer by a DMAC channel.

In **Figure 7.10**, both channels 0 and 1 are executing block transfer. At time tick 1, DMA transfer by channel 1 starts. At time tick 2, a DMA transfer request for channel 0 is accepted. As a result of DMAC channel arbitration, DMA transfer by channel 0 starts because channel 0 has a higher priority than channel 1. At time tick 3, the last transfer by channel 0 is complete, and the remaining DMA transfer in the block transfer by channel 1 starts. At time tick 4, the last transfer by channel 1 is complete. After time tick 5, DMA transfer by channel 0 and DMA transfer by channel 1 are executed similarly. At time tick 7, the DMA transfer by channel 0 is suspended and, as a result of DMAC channel arbitration, the DMA transfer by channel 1 starts. At time tick 8, the last transfer by channel 1 is complete, and then, at time tick 9, the DMA transfer by channel 0 resumes. At time tick 10, the last transfer by channel 0 is suspended again, and then, at time tick 11, the DMA transfer by channel 0 is aborted. At time tick 12, the suspended state for channel 0 is cleared, but DMA transfer is not executed because the DMA transfer is aborted at time tick 11.

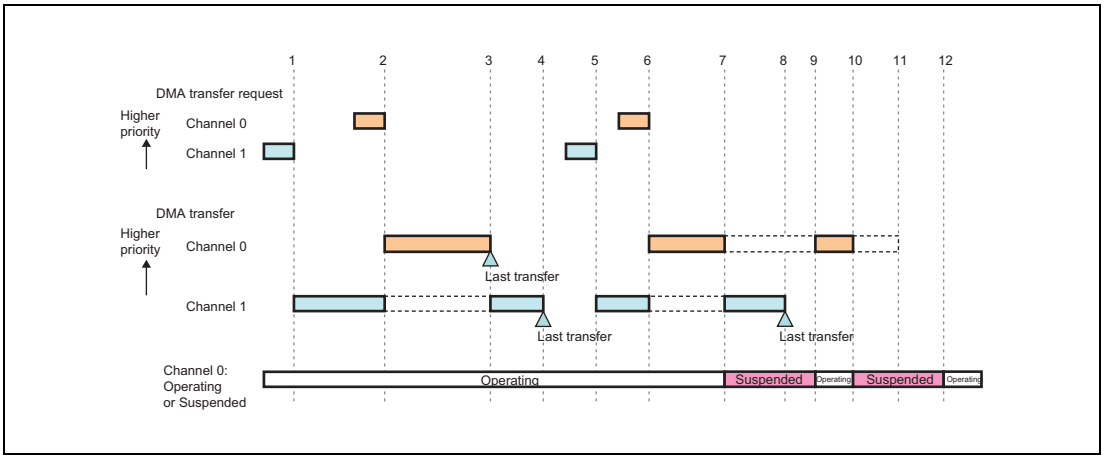


Figure 7.10 Example of Suspending, Resuming, and Aborting Transfer by a DMAC Channel

7.3.3 Suspending, Resuming, and Aborting Transfer by the DTS

You can suspend currently ongoing DMA transfer by the DTS by setting the DTS suspend bit in the DTS control register 1 (DTSCTL1.DTSUST) or the DMA suspend bit in the DMA control register (DMACTL.DMASPD). If a DMA cycle is ongoing, DMA transfer is suspended at the time when the DMA cycle is finished. If the ongoing DMA cycle is a single transfer or a transfer that completes a block transfer (the last transfer or address reload transfer), DMA transfer is suspended after a TI write back after the completion of the DMA cycle. If the ongoing DMA cycle is a type other than the above, the DMA transfer is suspended after the completion of the DMA cycle without a TI write back. To resume DMA transfer while it is suspended, clear the DTS suspend bit in the DTS control register 1 or the DMA suspend bit in the DMA control register.*1

Note 1. Before suspending or resuming currently ongoing DMA transfer by the DTS, the DMA transfer request enable bit (DTFSLnnn.REQEN) must be switched to disable and enable arbitration for the given channel. To suspend or resume DMA transfer, follow the procedure below.

(a) Procedure for suspending the DTS

1. Clear the DMA transfer request enable bit (DTFSLnnn.REQEN) for all DTS channels for which the DMA transfer request enable bit (DTFSLnnn.REQEN) has been set.
2. Set the DTSCTL1.DTSUST or DMACTL.DMASPD bit.

(b) Procedure for resuming the DTS

1. Set the DMA transfer request enable bit (DTFSLnnn.REQEN) for DTS channels, which was cleared in step 1 of (a).
2. Clear the DTSCTL1.DTSUST or DMACTL.DMASPD bit that was set in step 2 of (a).

If you want to abort currently ongoing DMA transfer by the DTS, suspend the DTS by using the DTS suspend bit in the DTS control register 1 (DTSCTL1.DTSUST), and then set the DTS transfer abort request bit in the DTS control register 2 (DTSCTL2.DTSTIT) to abort currently suspended DMA transfer. If transfer is aborted, no TI write back is executed. In addition, aborting DMA transfer does not change the value of the DTS suspend bit (DTSCTL1.DTSUST). If you want the DTS to accept another DMA transfer request after the transfer has been aborted, set the DMA transfer request enable bit (DTFSLnnn.REQEN) for DTS channels, and then clear the DTS suspend bit.

Figure 7.11 shows an example of suspending, resuming, and aborting transfer by the DTS.

In **Figure 7.11**, channels 0, 1, and 2 are executing block transfer. At time tick 1, a DMA transfer request for channel 1 is accepted and DMA transfer starts. At time tick 2, DMA transfer requests for channels 0 and 2 are generated. At time tick 3, the last transfer by channel 1 is complete, and as a result of DTS channel arbitration, a DMA transfer request for channel 0 is accepted, and DMA transfer by channel 0 starts because channel 0 has a higher priority than channel 2. At time tick 4, the last transfer by channel 0 is complete, and DMA transfer by channel 2 starts. At time tick 5, the DTS is put into the suspended state, and the DMA transfer by channel 2 is suspended. At time tick 6, DMA transfer requests for channels 0 and 1 are generated. At time tick 7, the suspended state for the DTS is cleared, and the DMA transfer by channel 2, which has been suspended in the middle of a block transfer, is resumed. If DMA transfer is suspended in the middle of a block transfer, no DTS channel arbitration is done when it is resumed. At time tick 8, the last transfer by channel 2 is complete, and as a result of DTS channel arbitration, a DMA transfer request for channel 0 is accepted and the DMA transfer starts because channel 0 has a higher priority than channel 1. At time tick 9, the DTS is put into the suspended state, and at time tick 10, the suspended DMA transfer by channel 0 is aborted. When the suspended state of the DTS 0 is cleared at time tick 11, DMA transfer by channel 1 starts because there is no currently ongoing DMA transfer and channel 1 is the only channel with a DMA transfer request.

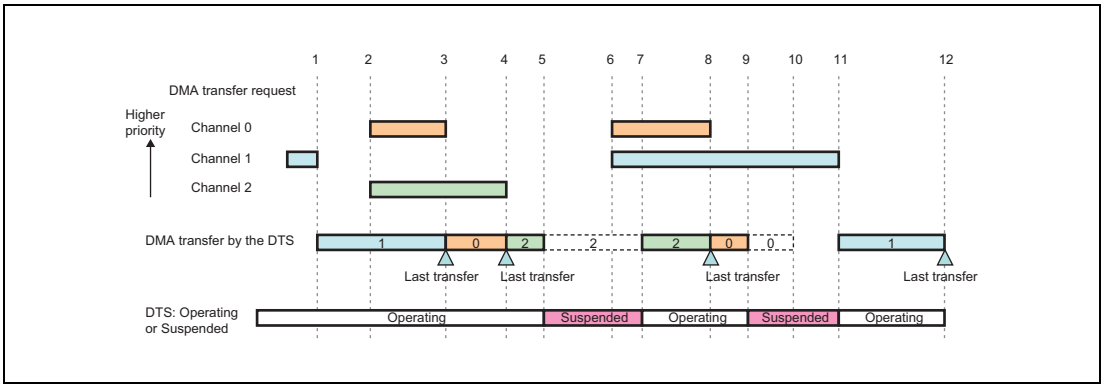


Figure 7.11 Example of Suspending, Resuming, and Aborting Transfer by the DTS

7.3.4 Masking and Clearing a Hardware DMA Transfer Request by the DTFR

If a DMAC uses a hardware DMA transfer request, you can temporarily disable (mask) the hardware DMA transfer request output from the DTFR to the DMAC by clearing the hardware DMA transfer source selection enable bit (DTFRn.REQEN) in the DTFR setting register.

Also, if a hardware DMA transfer source is used, you can clear a hardware DMA transfer request retained in the DTFR by using the hardware DMA transfer request clear (DTFRRQCn.DRQC) bit in the DTFR transfer request clear register.

Even if you suspend or abort DMA transfer by a DMAC channel, the hardware DMA transfer request selection/hold circuit of the DTFR is still running, and consequently, the DTFR may retain a hardware DMA transfer request that came to the DTFR during the suspension or transfer abort period of the DMAC channel. When you resume or start DMA transfer by a DMAC channel, clear the hardware DMA transfer request retained in the DTFR as required.

When a DMAC is used with hardware transfer request and block transfer 1 or 2, if the hardware transfer source selection enable bit is disabled (DTFRn.REQEN = 0), the ongoing block transfer is aborted.

7.3.5 Masking and Clearing a Hardware DMA Transfer Request by the DTSFSL

As for a DTS, you can temporarily disable (mask) a DMA transfer request from a channel to the DTS by clearing the transfer request enable bit (DTFSLnnn.REQEN) in the DTSFSL operation setting register. (The masking is actually done by excluding the channel from the candidates in DTS channel arbitration in the DTSFSL.)

Also, you can clear a DMA transfer request retained in the DTSFSL by using the transfer request clear (DTFSCnnn.DRQC) bit in the DTSFSL transfer request clear register.

Regardless of the state of the DTS and the transfer request enable bit (DTFSLnnn.REQEN) of the DTSFSL, the DTSFSL always monitors the hardware transfer source input, and a DMA transfer request for a channel is set when a hardware transfer source for the channel is input to the DTSFSL. When you resume or start DTS transfer, clear the hardware DMA transfer request retained in the DTSFSL as required.

7.3.6 List of Functions for Suspending, Resuming, and Aborting Transfer

Table 7.5 List of Functions for Suspending, Resuming, and Aborting Transfer

Function	How to execute the function	Operation	Possibility of aborting DMA transfer	Master that can execute the function (See Section 7.5, Reliability Function.)
Suspending and resuming DMA	Setting and clearing the DMACTL.DMASPD*2.	All channels are suspended.	Not possible*1	Special master
Suspending and resuming a DMAC channel	Clearing and setting the DCENn.DTE in each channel register.*3	DMA transfer by the channel is suspended.	Possible (by clearing the DMA transfer request flag during suspension)	Special master, and general master assigned to the channel.
Suspending and resuming the DTS	Setting and clearing the DTSCTL1.DTSUST*2.	DMA transfer by the DTS is suspended.	Possible (by setting the DTSCTL2.DTSTIT during suspension)	Special master

Note 1. In order to abort DMA transfer, you need to either abort transfer by the DMAC channel or abort transfer by the DTS.

Note 2. To suspend or resume currently ongoing DMA transfer by the DTS, the DMA transfer request enable bit (DTFSLnnn.REQEN) must be switched to disable and enable arbitration for the given channel. For details, refer to the notes in **Section 7.3.3, Suspending, Resuming, and Aborting Transfer by the DTS**.

Note 3. While continuous transfer is in use, set and clear the continuous transfer enable bit (DTCTn.MLE) first.

7.4 Error Control

7.4.1 Type of Error

DMA can generate the following two types of errors.

- DMA Transfer Error
  This error is generated when error is detected in the read cycle or write cycle in a DMA cycle.
  This error can be generated in all DMAC and DTS channels during execution of DMA transfer.
- DTSRAM Error
  This error is generated when ECC error is detected in the DTSRAM read access by a DTS.
  This error can be generated in the TI fetch during execution of DMA transfer by the DTS or while software is accessing the DTS channel registers.

7.4.2 DMA Transfer Error

A DMA transfer error leads to the generation of a DMA transfer error interrupt (DMAERR).

A DMA transfer error is detected independently for the DMAC channels and the DTS, but the DMA transfer error interrupt (DMAERR) is common to the DMAC and the DTS.

7.4.2.1 Operation of a DMAC When DMA Transfer Error Occurs

When DMA transfer error occurs in a DMAC, the transfer error flag (DCSTn.ER) in the DMAC transfer status register of the channel with the DMA transfer error. The DMAC error register (DMACER) shows the transfer error flags of all 16 DMAC channels.

While the transfer error flag of a channel is set, a new DMA cycle is not executed if the transfer error case DMA transfer disable setting (DTCTn.ESE) bit is set. On the other hand, a DMA cycle is executed regardless of the value of the transfer error flag if the transfer error case DMA transfer disable setting (DTCTn.ESE) bit is cleared.

If you want to abort the DMA transfer by the channel where a DMA transfer error occurred, follow the procedure to abort DMA transfer by the DMAC channel.

If DMA transfer error occurs during the read cycle of a DMA cycle, the write cycle is not executed. If DMA transfer error occurs during the write cycle of a DMA cycle, the validity of the write is not guaranteed.

Regardless of whether DMA transfer error occurs in the read cycle or write cycle of a DMA cycle, the source address, destination address, transfer count, and address reload count registers are updated.

7.4.2.2 Operation of a DTS When DMA Transfer Error Occurs

When DMA transfer error occurs in a DTS, the DTS error flag (DTSER1.DTSER) in the DTS error register is set, and the DTS channel number with the DMA transfer error is stored in the DTS error channel (DTSER1.DTSERCH) in the same register.

If DMA transfer error occurs in a single transfer, a TI write back is executed to finish the DMA cycle.

If DMA transfer error occurs in the middle of a block transfer and the transfer error case DMA transfer abort setting (DTTCTnnn.ESE) is set, the remaining DMA cycles in the block transfer are not executed, but a TI write back is executed to finish the DMA cycle. At the same time, the DTS transfer status (DTSSTS.DTSACT) bit in the DTS status register is cleared. If DMA transfer error occurs during a block transfer and the setting to abort the DMA transfer at the time of a transfer error (DTTCTnnn.ESE) is cleared, the block transfer continues regardless of the DMA transfer error.

If DMA transfer error occurs during the read cycle of a DMA cycle, the write cycle is not executed. If DMA transfer error occurs during the write cycle of a DMA cycle, the validity of the write is not guaranteed.

Regardless of whether DMA transfer error occurs in the read cycle or write cycle of a DMA cycle, the source address, destination address, transfer count, and address reload count registers are updated, and the TI is updated by a TI write back.

If the DTS error flag in the DTS error register is set, a TI fetch is executed when the DTS accepts a DMA transfer request for the channel with the same channel number as the one stored in the DTS error channel. If, as a result of the TI fetch, the setting to abort the DMA transfer at the time of a transfer error (DTTCTnnn.ESE) is found to be set, a DMA cycle and a TI write back are not executed. If the setting to abort the DMA transfer at the time of a transfer error (DTTCTnnn.ESE) is cleared, DMA transfer is executed.

If the DTS error flag in the DTS error register is set, DMA transfer is executed when the DTS accepts a DMA transfer request for a channel with a channel number other than the one stored in the DTS error channel.

7.4.3 DTSRAM Error

There are two types of DTSRAM errors detected in the DTSRAM read access: 1-bit ECC error and 2-bit ECC error.

If a 1-bit ECC error is detected during a TI fetch, error corrected data is used, and DMA transfer continues. If a 1-bit ECC error is detected during DTS channel register access from software, error corrected data is returned as read data. In either case, the DTSRAM 1-bit error flag (DTSER2.RAMSED) in the DTS error register 2 is set, and the address of the error location in the DTSRAM is stored to the DTSRAM 1-bit error address (DTSER2.RAMSEDAD).

In addition, the error is notified to the ECM.

If a 2-bit ECC error is detected during a TI fetch, handling of the DMA transfer request is terminated without executing a DMA cycle and TI write back. If a 2-bit ECC error is detected during DTS channel register access from software, peripheral bus error is notified. In either case, the DTSRAM 2 bit error flag (DTSER2.RAMDED) in the DTS error register 2 is set, and the address of the error location in the DTSRAM is stored to the DTSRAM 2 bit error address (DTSER2.RAMDEDAD). In addition, the error is notified to the ECM.

7.5 Reliability Function

7.5.1 Overview

In this product, DMA is a resource used by multiple masters (CPU1 or PCU). In order for DMA to support multi-core configuration, the following reliability functions are offered.

- Register access protection function
- Master information inherit function

7.5.2 Register Access Protection Function

This product is designed to assign each DMA channel to CPU1 or PCU.

The register access protection function allows access to the transfer information of each DMA channel from the master (CPU1 or PCU) assigned to the channel but prohibits access from other masters.

The register access protection function enables you, for example, to prevent the settings of the channel from being read or updated by masters other than the one assigned to the channel.

7.5.2.1 Identifying the Accessing Master

DMA identifies a master based on the ID of the accessing CPU (PEID), and whether the CPU is in supervisor mode (PSW.UM = 0) or user mode (PSW.UM = 1) .

7.5.2.2 Master Access

Master access includes the following two types.

- Special master access (CPU1 supervisor mode (UM = 0))
- General master access (accesses other than special master access)

In special master access, access to all registers is allowed.

In general master access, access to the following registers is allowed.

- The following global registers

  DMACER, DTSER1, DTSER2, and DTSSTS
- Channel registers of the channels assigned by the channel assignment. (For details, see **Section 7.5.2.3, Channel Assignment**.)

In general master access, access to registers other than the above is not allowed.

7.5.2.3 Channel Assignment

To each channel, DMA can assign a master (CPU1 or PCU) so that the master is allowed to use the channel. Channel assignment is configured in the channel master setting register (DMnnCM in the case of a DMAC and DTSnnnCM in the case of a DTS) in supervisor mode (UM = 0) by CPU1.

In general master access, the master assigned to a channel by channel assignment is allowed to access the channel registers of the channel. If the channel registers of a channel is accessed by a master other than the master assigned to the channel, the access is called illegal access. For information about illegal access, see **Section 7.5.2.4, Illegal Access**.

7.5.2.4 Illegal Access

DMA handles the following access as illegal access.

(a) General master access to the global registers
Except the following registers: DMACER, DTSER1, DTSER2, and DTSSTS

(b) General master access to the channel registers of a channel by a master other than the master assigned to the channel

DMA's actions against illegal access are as follows.

For both cases (a) and (b),

- Write access is ignored.
- Read access returns 0 as read data.

Only for the case (b),

- The information about the illegal access is stored in a register access protection violation register. The DMAC0, DMAC1, and DTS have their own register access protection violation registers (DM0CMV, DM1CMV, and DTSCMV respectively).
- The ECM is notified of DMAVIOL (illegal DMA access).

Only the special master can access the register access protection violation registers. The special master can check whether illegal access has occurred by checking the register access protection violation registers periodically.

In addition, it is recommended that, when a master tries to use DMA and configures transfer information in the channel registers, the master should check whether the configuration has been successfully completed without illegal access by, for example, reading back the settings.

7.5.3 Master Information Inherit Function

In this product, DMA access inherits master information that is equivalent to the master information of the CPU1 or PCU assigned to the DMA channel.

The master information that is output from DMA is as in **Table 7.6**.

Table 7.6 Master Information That Is Output from DMA

Meaning	Value that is output from DMA
UM	Same as the UM bit value in the channel master setting register
SPID	Same as the SPID bit value in the channel master setting register
PEID	Same as the PEID bit value in the channel master setting register
DMA access	1

7.5.4 Other Reliability Functions

7.5.4.1 Restriction on the Next Channel in the Chain

The reliability function limits the channels you can select as the next channel in the chain.

When you use the chain function, the channel master settings of a channel and its next channel in the chain must be the same.

The chain function is designed so that a channel and its next channel in the chain are managed by the same master.

When DMA detects that different masters are assigned to a channel and its next channel in the chain, it is deemed illegal and the chain function is suppressed. More specifically, when DMA tries to execute the chain function, DMA compares the chain master settings of the channel and its next channel in the chain, and if the settings are the same for PEID and UM, the chain function is allowed and a chain request is sent to the next channel. If the settings are not the same for PEID and UM, a chain request is not sent.

7.6 Setting Up DMA Transfer

7.6.1 Overview of Setting Up DMA

Table 7.7 Channel Assignment (1/2)

No.	Master that configures the setting	Description	Register		Necessity of the setting
1	Special master (supervisor mode (UM = 0) of CPU1)	Overall DMA operation setting	DTSPR0 to DTSPR7	DTS channel priority setting register	Mandatory (if a DTS is used)
2			DM00CM to DM17CM	DMAC channel master setting register	Mandatory (if a DMAC is used)
3			DTS0CM to DTS127CM	DTS channel master setting register	Mandatory (if a DTS is used)
4		Status clear	DTSERC	DTS error clear register	Recommended
5			CMVC	Channel protection violation clear register	Recommended
6	Master assigned to the DMAC channel	Channel setting	DSAn	DMAC source address register	Mandatory
7			DDAn	DMAC destination address register	Mandatory
8			DTCn	DMAC transfer count register	Mandatory
9			DTCTn	DMAC transfer control register	Mandatory
10			DRSAn	DMAC reload source address register	Mandatory if the reload function is used
11			DRDAn	DMAC reload destination address register	Mandatory if the reload function is used
12			DRTCn	DMAC reload transfer count register	Mandatory if the reload function is used
13			DTCCn	DMAC transfer count compare register	Mandatory if the transfer count match interrupt is used
14			DTFRn	DTFR setting register	Mandatory
15		Status clear	DCSTCn	DMAC transfer status clear register	Mandatory
16			DTFRRQCn	DTFR transfer request clear register	Recommended
17		Channel operation enable	DCENn	DMAC channel operation enable setting register	Mandatory

Table 7.7 Channel Assignment (2/2)

No.	Master that configures the setting	Description	Register		Necessity of the setting
18	Master assigned to the DTS channel	Channel setting	DTSAnnn	DTS source address register	Mandatory
19			DTDAnnn	DTS destination address register	Mandatory
20			DTTCnnn	DTS transfer count register	Mandatory
21			DTTCTnnn	DTS transfer control register	Mandatory
22			DTRSAnnn	DTS reload source address register	Mandatory if the reload function is used
23			DTRDAnnn	DTS reload destination address register	Mandatory if the reload function is used
24			DTRTCnnn	DTS reload transfer count register	Mandatory if the reload function is used
25			DTTCCnnn	DTS transfer count compare register	Mandatory if the transfer count match interrupt is used
26		Status clear	DTFSCnnn	DTSFSL transfer request clear register	Recommended
27		Transfer request enable	DTFSLnnn	DTSFSL operation setting register	Mandatory

7.6.2 Setting Up the Overall DMA Operation

You need to set up the overall DMA operation before you start using DMA.

To configure the overall DMA operation, the special master (supervisor mode (UM = 0) of CPU1) needs to set up global registers. Global registers can be set up only by special master access. For details, see **Section 7.5, Reliability Function**.

The following registers must be set up to configure the overall DMA operation.

- DTS channel priority setting registers (DTSPRn, n = 0 to 7)
  Those registers configure the priority level of each DTS channel used for DTS channel arbitration.
- DMAC channel master setting registers (DMnnCM)
- DTS channel master setting registers (DTSnnnCM)
  Those registers configure channel assignment. (For details, see **Section 7.5, Reliability Function**.)
  If the DMAC channel master setting registers and the DTS channel master setting registers are not properly set, DMA channel setting and DMA transfer cannot be executed properly.

Also, if errors are detected in the following registers while the overall DMA operation is set up, clearing the errors is recommended.

- DTS error register 1 (DTSER1)
- DTS error register 2 (DTSER2)
- DMAC0 register access protection violation register (DM0CMV)
- DMAC1 register access protection violation register (DM1CMV)
- DTS register access protection violation register (DTSCMV)

7.6.3 Setting Up the DMA Channel Setting

The DMA channel setting defines the transfer information and transfer source for each DMAC and DTS channel.

To configure the DMA channel setting, the master assigned to each channel by the channel assignment needs to set up channel registers.

7.6.3.1 Setting Up the DMAC Channel Setting

Follow the procedure below to set up the DMAC channel setting and use the DMAC.

(1) Disabling the DMAC Channel Operation

If the channel operation enable (DTE) in the DMAC channel operation enable setting register (DCENn) is set, clear the DTE bit to disable the channel operation.

(2) Setting Up the Transfer Information

When you set up the transfer information of the DMAC, the following registers need to be set up.

- DMAC source address register (DSAn)
- DMAC destination address register (DDAn)
- DMAC transfer count register (DTCn)
- DMAC transfer control register (DTCTn)
- DMAC reload source address register (DRSAn)
- DMAC reload destination address register (DRDAn)
- DMAC reload transfer count register (DRTCn)
- DMAC transfer count compare register (DTCCn)

(3) Setting Up the DMA Transfer Request

While setting the transfer information, you need to set up the DMA transfer request selection assignment (DTCTn.DRS) bit in the DMAC transfer control register (DTCTn) to define whether the hardware or software DMA transfer request is used.

You cannot use both the hardware and software DMA transfer requests for the same channel at the same time.

If you use the hardware DMA transfer request, you need to select one source used as the hardware DMA transfer request out of 128 hardware DMA transfer sources using the hardware DMA transfer source selection (DTFRn.REQSEL) in the DTFR setting register. Also, you need to enable the hardware DMA transfer source selection (DTFRn.REQEN) in the same register.

The DTFR may retain a hardware DMA transfer request that came before the hardware DMA transfer source is selected. Clear the hardware DMA transfer request (DTFRRQn.DRQ) retained in the DTFR using the DTFR transfer request clear register (DTFRRQCn) if necessary.

If you use the software DMA transfer request, disable the hardware DMA transfer source selection (DTFRn.REQEN) in the DTFR setting register.

(4) Clearing the Transfer Status

The DMAC transfer status register (DCSTn) may retain the result of the previous DMA transfer. You need to clear the flags in the DMAC transfer status register using the DMAC transfer status clear register (DCSTCn).

(5) Enabling the DMAC Channel Operation

Set the channel operation enable (DCENn.DTE) bit in the DMAC channel operation enable setting register to enable the channel operation.

After the channel operation enable bit is set, the DMAC can accept a DMA transfer request and start DMA transfer.

7.6.3.2 Setting Up the DTS Channel Setting

Follow the procedure below to set up the DTS channel setting and use the DTS.

(1) Disabling the Transfer Request by the DTSFSL

Clear the transfer request enable (DTFSLnnn.REQEN) bit in the DTSFSL operation setting register of the DTS channel you want to set up the channel setting for. This procedure is not mandatory but recommended in order to prevent a DMA transfer request from being sent mistakenly to the DTS channel currently being configured.

It is also recommended to check the DTS status register (DTSSTS) and confirm that DMA transfer is not ongoing for the DTS channel currently being configured.

(2) Setting Up the Transfer Information

When you set up the transfer information of the DTS, the following registers need to be set up to configure the transfer information.

- DTS source address register (DTSAnnn)
- DTS destination address register (DTDAnnn)
- DTS transfer count register (DTTCnnn)
- DTS transfer control register (DTTCTnnn)
- DTS reload source address register (DTRSAnnn)
- DTS reload destination address register (DTRDAnnn)
- DTS reload transfer count register (DTRTCnnn)
- DTS transfer count compare register (DTTCCnnn)

(3) Setting Up the DMA Transfer Request

Unlike a DMAC, a DTS does not care whether a DMA transfer request is a hardware DMA transfer request or software DMA transfer request. A DTS has a transfer request pending bit for each channel in the DTSFSL, and both a hardware and software DMA transfer requests are retained in the same transfer request pending bit (DTFSTnnn.DRQ). Therefore, a DTS has no setting for selecting whether the hardware or software transfer request is used.

The DTSFSL may retain a DMA transfer request that came before the transfer information is set up. Clear the DMA transfer request (DTFSTnnn.DRQ) retained in the DTSFSL if necessary, using the DTSFSL transfer request clear register (DTFSCnnn).

(4) Enabling the Transfer Request by the DTSFSL

Set the transfer request enable (DTFSLnnn.REQEN) bit in the DTSFSL operation setting register to enable the DMA transfer request for the DTS channel.

After the transfer request enable bit for the DTSFSL is set, the DTS can accept a DMA transfer request and start DMA transfer.

7.7 DMA Trigger Source

7.7.1 List of DMA Trigger Sources

The DMA trigger source assignment for DMA channel n is set in the DTFR setting register (DTFRn).

Table 7.8 List of DMA Trigger Sources (1/4)

	Shared with DTSTRG	Function/Module	DMA Trigger Source
DMACTRG[0]	—	ATUIV TimerD	OCR2D00 compare match interrupt
DMACTRG[1]	—		OCR2D01 compare match interrupt
DMACTRG[2]	—		OCR2D02 compare match interrupt
DMACTRG[3]	—		OCR2D03 compare match interrupt
DMACTRG[4]	—		DCNTD00 down counter underflow interrupt
DMACTRG[5]	—		DCNTD01 down counter underflow interrupt
DMACTRG[6]	—		DCNTD02 down counter underflow interrupt
DMACTRG[7]	—		DCNTD03 down counter underflow interrupt
DMACTRG[8]	—		OCR2D10 compare match interrupt
DMACTRG[9]	—		OCR2D11 compare match interrupt
DMACTRG[10]	—		OCR2D12 compare match interrupt
DMACTRG[11]	—		OCR2D13 compare match interrupt
DMACTRG[12]	—		DCNTD10 down counter underflow interrupt
DMACTRG[13]	—		DCNTD11 down counter underflow interrupt
DMACTRG[14]	—		DCNTD12 down counter underflow interrupt
DMACTRG[15]	—		DCNTD13 down counter underflow interrupt
DMACTRG[16]	—		OCR2D20 compare match interrupt
DMACTRG[17]	—		OCR2D21 compare match interrupt
DMACTRG[18]	—		OCR2D22 compare match interrupt
DMACTRG[19]	—		OCR2D23 compare match interrupt
DMACTRG[20]	—		DCNTD20 down counter underflow interrupt
DMACTRG[21]	—		DCNTD21 down counter underflow interrupt
DMACTRG[22]	—		DCNTD22 down counter underflow interrupt
DMACTRG[23]	—		DCNTD23 down counter underflow interrupt
DMACTRG[24]	—		DCNTD30 down counter underflow interrupt
DMACTRG[25]	—		DCNTD31 down counter underflow interrupt
DMACTRG[26]	—		DCNTD32 down counter underflow interrupt
DMACTRG[27]	—		DCNTD33 down counter underflow interrupt
DMACTRG[28]	—		DCNTD40 down counter underflow interrupt
DMACTRG[29]	—		DCNTD41 down counter underflow interrupt
DMACTRG[30]	—		DCNTD42 down counter underflow interrupt
DMACTRG[31]	—		DCNTD43 down counter underflow interrupt
DMACTRG[32]	—		OCR2D40 compare match interrupt
DMACTRG[33]	—		OCR2D41 compare match interrupt
DMACTRG[34]	—		OCR2D42 compare match interrupt
DMACTRG[35]	—		OCR2D43 compare match interrupt
DMACTRG[36]	—		OCR2D60 compare match interrupt

Table 7.8 List of DMA Trigger Sources (2/4)

	Shared with DTSTRG	Function/Module	DMA Trigger Source
DMACTRG[37]	—	ATUIV TimerD	OCR2D61 compare match interrupt
DMACTRG[38]	—		OCR2D62 compare match interrupt
DMACTRG[39]	—		OCR2D63 compare match interrupt
DMACTRG[40]	—		OCR2D90 compare match interrupt
DMACTRG[41]	—		OCR2D91 compare match interrupt
DMACTRG[42]	—		OCR2D92 compare match interrupt
DMACTRG[43]	—		OCR2D93 compare match interrupt
DMACTRG[44]	—	ATUIV TimerE	CYLRE00 cycle compare match DMA startup request
DMACTRG[45]	—		CYLRE10 cycle compare match DMA startup request
DMACTRG[46]	—		CYLRE20 cycle compare match DMA startup request
DMACTRG[47]	—		CYLRE30 cycle compare match DMA startup request
DMACTRG[48]	—		CYLRE40 cycle compare match DMA startup request
DMACTRG[49]	—		CYLRE50 cycle compare match DMA startup request
DMACTRG[50]	—		CYLRE60 cycle compare match DMA startup request
DMACTRG[51]	DTSTRG[70]	ATUIV TimerF	TimerF0 input capture interrupt
DMACTRG[52]	DTSTRG[71]		TimerF1 input capture interrupt
DMACTRG[53]	DTSTRG[72]		TimerF2 input capture interrupt
DMACTRG[54]	DTSTRG[73]		TimerF3 input capture interrupt
DMACTRG[55]	DTSTRG[74]		TimerF4 input capture interrupt
DMACTRG[56]	DTSTRG[75]		TimerF5 input capture interrupt
DMACTRG[57]	DTSTRG[76]		TimerF6 input capture interrupt
DMACTRG[58]	DTSTRG[77]		TimerF7 input capture interrupt
DMACTRG[59]	—		TimerF8 input capture interrupt
DMACTRG[60]	—		TimerF9 input capture interrupt
DMACTRG[61]	—		TimerF10 input capture interrupt
DMACTRG[62]	—		TimerF11 input capture interrupt
DMACTRG[63]	—		TimerF12 input capture interrupt
DMACTRG[64]	—		TimerF13 input capture interrupt
DMACTRG[65]	—		TimerF14 input capture interrupt
DMACTRG[66]	—		TimerF15 input capture interrupt
DMACTRG[67]	—	ATUIV TimerG	OCRG0 compare match interrupt
DMACTRG[68]	—		OCRG1 compare match interrupt
DMACTRG[69]	—		OCRG2 compare match interrupt
DMACTRG[70]	—		OCRG3 compare match interrupt
DMACTRG[71]	—		OCRG4 compare match interrupt
DMACTRG[72]	—		OCRG5 compare match interrupt
DMACTRG[73]	—		OCRG6 compare match interrupt
DMACTRG[74]	—		OCRG7 compare match interrupt
DMACTRG[75]	—	Reserved	—
DMACTRG[76]	—	Reserved	—
DMACTRG[77]	—	Reserved	—
DMACTRG[78]	—	Reserved	—

Table 7.8 List of DMA Trigger Sources (3/4)

	Shared with DTSTRG	Function/Module	DMA Trigger Source
DMACTRG[79]	—	APA	PWM output ch.0 status modify interrupt
DMACTRG[80]	—		PWM output ch.1 status modify interrupt
DMACTRG[81]	—		PWM output ch.2 status modify interrupt
DMACTRG[82]	—		PWM output ch.3 status modify interrupt
DMACTRG[83]	—		PWM output ch.4 status modify interrupt
DMACTRG[84]	—		PWM output ch.5 status modify interrupt
DMACTRG[85]	—		PWM output ch.6 status modify interrupt
DMACTRG[86]	—		PWM output ch.7 status modify interrupt
DMACTRG[87]	—		PWM output ch.8 status modify interrupt
DMACTRG[88]	—		PWM output ch.9 status modify interrupt
DMACTRG[89]	—		PWM output ch.10 status modify interrupt
DMACTRG[90]	—		PWM output ch.11 status modify interrupt
DMACTRG[91]	—		PWM output ch.12 status modify interrupt
DMACTRG[92]	—		PWM output ch.13 status modify interrupt
DMACTRG[93]	—		PWM output ch.14 status modify interrupt
DMACTRG[94]	—		PWM output ch.15 status modify interrupt
DMACTRG[95]	—	DS-ADC	DSADI0 DSADC0 A/D conversion completion interrupt (DMA request)
DMACTRG[96]	—		DSADI1 DSADC1 A/D conversion completion interrupt (DMA request)
DMACTRG[97]	—		DSADI2 DSADC2 A/D conversion completion interrupt (DMA request)
DMACTRG[98]	—		DSADI3 DSADC3 A/D conversion completion interrupt (DMA request)
DMACTRG[99]	—		DSADI4 DSADC4 A/D conversion completion interrupt (DMA request)
DMACTRG[100]	—		DSADI5 DSADC5 A/D conversion completion interrupt (DMA request)
DMACTRG[101]	—		DSADI6 DSADC6 A/D conversion completion interrupt (DMA request)
DMACTRG[102]	—		DSADI7 DSADC7 A/D conversion completion interrupt (DMA request)
DMACTRG[103]	DTSTRG[90]	RHSB_0	Downstream data DMA request
DMACTRG[104]	DTSTRG[91]		Upstream DMA request
DMACTRG[105]	DTSTRG[93]	RHSB_1	Downstream data DMA request
DMACTRG[106]	DTSTRG[94]		Upstream DMA request
DMACTRG[107]	—	SCI_0	RXI (receive data full)
DMACTRG[108]	—		TXI (transmit data empty)
DMACTRG[109]	—	SCI_1	RXI (receive data full)
DMACTRG[110]	—		TXI (transmit data empty)
DMACTRG[111]	—	SCI_2	RXI (receive data full)
DMACTRG[112]	—		TXI (transmit data empty)
DMACTRG[113]	—	SCI_3	RXI (receive data full)
DMACTRG[114]	—		TXI (transmit data empty)
DMACTRG[115]	—	Reserved	—
DMACTRG[116]	—	Reserved	—
DMACTRG[117]	—	External interrupt	IRQ0 interrupt*2
DMACTRG[118]	—		IRQ1 interrupt*2
DMACTRG[119]	—		IRQ2 interrupt*2
DMACTRG[120]	—		IRQ3 interrupt*2
DMACTRG[121]	—	Reserved	—

Table 7.8 List of DMA Trigger Sources (4/4)

	Shared with DTSTRG	Function/Module	DMA Trigger Source
DMACTRG[122]	—	Reserved	—
DMACTRG[123]	—	Reserved	—
DMACTRG[124]	—	Reserved	—
DMACTRG[125]	—	Data flash	Signal for DMA program command*3
DMACTRG[126]	—	Reserved	—
DMACTRG[127]	—	Reserved	—

Note 1. The name of DMA trigger source may be different from that of output signal in each module.

Note 2. When the IRQ interrupt is used as the DMA source, set the external interrupt detection method select bit in the external interrupt control register (EXINTCTL) to high level (01_H).

Note 3. See ***RH850/E1x Flash Memory User's Manual Hardware Interface***.

7.8 DTS Trigger Source

7.8.1 List of DTS Trigger Sources

Table 7.9 shows the DTS trigger source assignment for DTS channel n.

Table 7.9 List of DTS Trigger Sources (1/4)

	Shared with DMACTRG	Function/Module	DMA Trigger Source
DTSTRG[0]	—	OSTM	OSTM0 interrupt (OSTM0TINT)
DTSTRG[1]	—		OSTM1 interrupt (OSTM1TINT)
DTSTRG[2]	—		OSTM2 interrupt (OSTM2TINT)
DTSTRG[3]	—	Reserved	—
DTSTRG[4]	—	ADC	ADI00 ADC0 scan group 0 completion interrupt
DTSTRG[5]	—		ADI01 ADC0 scan group 1 completion interrupt
DTSTRG[6]	—		ADI02 ADC0 scan group 2 completion interrupt
DTSTRG[7]	—		ADI03 ADC0 scan group 3 completion interrupt
DTSTRG[8]	—		ADI04 ADC0 scan group 4 completion interrupt
DTSTRG[9]	—		ADI10 ADC1 scan group 0 completion interrupt
DTSTRG[10]	—		ADI11 ADC1 scan group 1 completion interrupt
DTSTRG[11]	—		ADI12 ADC1 scan group 2 completion interrupt
DTSTRG[12]	—		ADI13 ADC1 scan group 3 completion interrupt
DTSTRG[13]	—		ADI14 ADC1 scan group 4 completion interrupt
DTSTRG[14]	—		ADC0 MPX interrupt (ADMPXI0)
DTSTRG[15]	—		ADC1 MPX interrupt (ADMPXI1)
DTSTRG[16]	—	ASF (AD addition function)	ASI0 ch0 integration completion interrupt
DTSTRG[17]	—		ASI1 ch1 integration completion interrupt
DTSTRG[18]	—		ASI2 ch2 integration completion interrupt
DTSTRG[19]	—		ASI3 ch3 integration completion interrupt
DTSTRG[20]	—		ASI4 ch4 integration completion interrupt
DTSTRG[21]	—		ASI5 ch5 integration completion interrupt
DTSTRG[22]	—	DFE	CH0 output data interrupt / CH0 condition match interrupt
DTSTRG[23]	—		CH1 output data interrupt / CH1 condition match interrupt
DTSTRG[24]	—		CH2 output data interrupt / CH2 condition match interrupt
DTSTRG[25]	—		CH3 output data interrupt / CH3 condition match interrupt
DTSTRG[26]	—		CH4 output data interrupt / CH4 condition match interrupt
DTSTRG[27]	—		CH5 output data interrupt / CH5 condition match interrupt
DTSTRG[28]	—		CH6 output data interrupt / CH6 condition match interrupt
DTSTRG[29]	—		CH7 output data interrupt / CH7 condition match interrupt
DTSTRG[30]	—		CH8 output data interrupt / CH8 condition match interrupt
DTSTRG[31]	—		CH9 output data interrupt / CH9 condition match interrupt
DTSTRG[32]	—		CH10 output data interrupt
DTSTRG[33]	—		CH11 output data interrupt
DTSTRG[34]	—		CH12 output data interrupt
DTSTRG[35]	—		CH13 output data interrupt
DTSTRG[36]	—		CH14 output data interrupt
DTSTRG[37]	—		CH15 output data interrupt

Table 7.9 List of DTS Trigger Sources (2/4)

	Shared with DMACTRG	Function/Module	DMA Trigger Source
DTSTRG[38]	—	ATU IV TimerA	ICRA0 input capture interrupt
DTSTRG[39]	—		ICRA1 input capture interrupt
DTSTRG[40]	—		ICRA2 input capture interrupt
DTSTRG[41]	—		ICRA3 input capture interrupt
DTSTRG[42]	—		ICRA4 input capture interrupt
DTSTRG[43]	—		ICRA5 input capture interrupt
DTSTRG[44]	—		ICRA6 input capture interrupt
DTSTRG[45]	—	ATU IV TimerB	OCRB12 compare match interrupt
DTSTRG[46]	—		ICRB0 input capture interrupt
DTSTRG[47]	—	ATU IV TimerC	GRC00 input capture/compare match interrupt
DTSTRG[48]	—		GRC01 input capture/compare match interrupt
DTSTRG[49]	—		GRC02 input capture/compare match interrupt
DTSTRG[50]	—		GRC03 input capture/compare match interrupt
DTSTRG[51]	—		GRC10 input capture/compare match interrupt
DTSTRG[52]	—		GRC11 input capture/compare match interrupt
DTSTRG[53]	—		GRC12 input capture/compare match interrupt
DTSTRG[54]	—		GRC13 input capture/compare match interrupt
DTSTRG[55]	—		GRC20 input capture/compare match interrupt
DTSTRG[56]	—		GRC21 input capture/compare match interrupt
DTSTRG[57]	—		GRC22 input capture/compare match interrupt
DTSTRG[58]	—		GRC23 input capture/compare match interrupt
DTSTRG[59]	—		GRC30 input capture/compare match interrupt
DTSTRG[60]	—		GRC31 input capture/compare match interrupt
DTSTRG[61]	—		GRC32 input capture/compare match interrupt
DTSTRG[62]	—		GRC33 input capture/compare match interrupt
DTSTRG[63]	—		GRC40 input capture/compare match interrupt
DTSTRG[64]	—		GRC41 input capture/compare match interrupt
DTSTRG[65]	—		GRC42 input capture/compare match interrupt
DTSTRG[66]	—		GRC43 input capture/compare match interrupt
DTSTRG[67]	—		GRC50 input capture/compare match interrupt
DTSTRG[68]	—		GRC60 input capture/compare match interrupt
DTSTRG[69]	—		GRC70 input capture/compare match interrupt
DTSTRG[70]	DMACTRG[51]	ATU IV TimerF	TimerF0 input capture interrupt
DTSTRG[71]	DMACTRG[52]		TimerF1 input capture interrupt
DTSTRG[72]	DMACTRG[53]		TimerF2 input capture interrupt
DTSTRG[73]	DMACTRG[54]		TimerF3 input capture interrupt
DTSTRG[74]	DMACTRG[55]		TimerF4 input capture interrupt
DTSTRG[75]	DMACTRG[56]		TimerF5 input capture interrupt
DTSTRG[76]	DMACTRG[57]		TimerF6 input capture interrupt
DTSTRG[77]	DMACTRG[58]		TimerF7 input capture interrupt

Table 7.9 List of DTS Trigger Sources (3/4)

	Shared with DMACTRG	Function/Module	DMA Trigger Source
DTSTRG[78]	—	ATUIV TimerJ	FIFOJ0 FIFO data full interrupt
DTSTRG[79]	—		FIFOJ1 FIFO data full interrupt
DTSTRG[80]	—		FIFOJ2 FIFO data full interrupt
DTSTRG[81]	—		FIFOJ3 FIFO data full interrupt
DTSTRG[82]	—		FIFOJ4 FIFO data full interrupt
DTSTRG[83]	—		FIFOJ5 FIFO data full interrupt
DTSTRG[84]	—	Reserved	—
DTSTRG[85]	—	Reserved	—
DTSTRG[86]	—	Reserved	—
DTSTRG[87]	—	Reserved	—
DTSTRG[88]	—	Reserved	—
DTSTRG[89]	—	Reserved	—
DTSTRG[90]	DMACTRG[103]	RHSB_0	Downstream data DMA request
DTSTRG[91]	DMACTRG[104]		Upstream DMA request
DTSTRG[92]	—		Downstream command DMA request
DTSTRG[93]	DMACTRG[105]	RHSB_1	Downstream data DMA request
DTSTRG[94]	DMACTRG[106]		Upstream DMA request
DTSTRG[95]	—		Downstream command DMA request
DTSTRG[96]	—	TSG2_0	TSG20 compare match interrupt 11 (INTTSG20I11)
DTSTRG[97]	—		TSG20 compare match interrupt 12 (INTTSG20I12)
DTSTRG[98]	—		TSG20 peak interrupt (INTTSG20IPEK)
DTSTRG[99]	—		TSG20 trough interrupt (INTTSG20IVLY)
DTSTRG[100]	—	TSG2_1	TSG21 compare match interrupt 11 (INTTSG21I11)
DTSTRG[101]	—		TSG21 compare match interrupt 12 (INTTSG21I12)
DTSTRG[102]	—		TSG21 peak interrupt (INTTSG21IPEK)
DTSTRG[103]	—		TSG21 trough interrupt (INTTSG21IVLY)
DTSTRG[104]	—	CSIH_0	Communication status interrupt (INT_CSIHTIC)
DTSTRG[105]	—		Receive status interrupt (INT_CSIHTIR)
DTSTRG[106]	—		Job completion interrupt (INT_CSIHTIJC)
DTSTRG[107]	—	CSIH_1	Communication status interrupt (INT_CSIHTIC)
DTSTRG[108]	—		Receive status interrupt (INT_CSIHTIR)
DTSTRG[109]	—		Job completion interrupt (INT_CSIHTIJC)
DTSTRG[110]	—	CSIH_2	Communication status interrupt (INT_CSIHTIC)
DTSTRG[111]	—		Receive status interrupt (INT_CSIHTIR)
DTSTRG[112]	—		Job completion interrupt (INT_CSIHTIJC)
DTSTRG[113]	—	CSIH_3	Communication status interrupt (INT_CSIHTIC)
DTSTRG[114]	—		Receive status interrupt (INT_CSIHTIR)
DTSTRG[115]	—		Job completion interrupt (INT_CSIHTIJC)
DTSTRG[116]	—	Reserved	—
DTSTRG[117]	—	Reserved	—
DTSTRG[118]	—	Reserved	—
DTSTRG[119]	—	Reserved	—
DTSTRG[120]	—	Reserved	—

Table 7.9 List of DTS Trigger Sources (4/4)

	Shared with DMACTRG	Function/Module	DMA Trigger Source
DTSTRG[121]	—	Reserved	—
DTSTRG[122]	—	Reserved	—
DTSTRG[123]	—	Reserved	—
DTSTRG[124]	—	Reserved	—
DTSTRG[125]	—	Reserved	—
DTSTRG[126]	—	Reserved	—
DTSTRG[127]	—	Reserved	—

7.9 Global Register

7.9.1 List of Global Register Addresses

Address = Base address "FFFF 8000_H" + Offset address

Table 7.10 List of Global Register Addresses (1/2)

Offset Address	Register Symbol	Meaning	Accessed Permission	
			Special Master	General Master
0000_H	DMACTL	DMA control register	√	×
0010_H	DTSCTL1	DTS control register 1	√	×
0014_H	DTSCTL2	DTS control register 2	√	×
0018_H	DTSSTS	DTS status register	√	√
0020_H	DMACER	DMAC error register	√	√
0024_H	DTSER1	DTS error register 1	√	√
0028_H	DTSER2	DTS error register 2	√	√
$002C_H$	DTSERC	DTS error clear register	√	×
0030_H	DM0CMV	DMAC0 register access protection violation register	√	×
0034_H	DM1CMV	DMAC1 register access protection violation register	√	×
0038_H	DTSCMV	DTS register access protection violation register	√	×
$003C_H$	CMVC	Register access protection violation clear register	√	×
0060_H	DTSPR0	DTS channel priority setting register 0	√	×
0064_H	DTSPR1	DTS channel priority setting register 1	√	×
0068_H	DTSPR2	DTS channel priority setting register 2	√	×
$006C_H$	DTSPR3	DTS channel priority setting register 3	√	×
0070_H	DTSPR4	DTS channel priority setting register 4	√	×
0074_H	DTSPR5	DTS channel priority setting register 5	√	×
0078_H	DTSPR6	DTS channel priority setting register 6	√	×
$007C_H$	DTSPR7	DTS channel priority setting register 7	√	×
0080_H	DTRECCTL	DTSRAM ECC control register	√	×
0084_H	DTRERINT	DTSRAM Error notification control register	√	×
0094_H	DTRTSCTL	DTSRAM test control register	√	×
0098_H	DTRTWDAT	DTSRAM test write data register	√	×
$009C_H$	DTRTRDAT	DTSRAM test read data register	√	×
0100_H	DM00CM	DMAC channel 0 channel master setting register	√	×
0104_H	DM01CM	DMAC0 channel 1 channel master setting register	√	×
0108_H	DM02CM	DMAC0 channel 2 channel master setting register	√	×
$010C_H$	DM03CM	DMAC0 channel 3 channel master setting register	√	×
0110_H	DM04CM	DMAC0 channel 4 channel master setting register	√	×
0114_H	DM05CM	DMAC0 channel 5 channel master setting register	√	×
0118_H	DM06CM	DMAC0 channel 6 channel master setting register	√	×
$011C_H$	DM07CM	DMAC0 channel 7 channel master setting register	√	×
0120_H	DM10CM	DMAC1 channel 0 channel master setting register	√	×
0124_H	DM11CM	DMAC1 channel 1 channel master setting register	√	×
0128_H	DM12CM	DMAC1 channel 2 channel master setting register	√	×
$012C_H$	DM13CM	DMAC1 channel 3 channel master setting register	√	×
0130_H	DM14CM	DMAC1 channel 4 channel master setting register	√	×

Table 7.10 List of Global Register Addresses (2/2)

Offset Address	Register Symbol	Meaning	Accessed Permission	
			Special Master	General Master
0134_H	DM15CM	DMAC1 channel 5 channel master setting register	√	×
0138_H	DM16CM	DMAC1 channel 6 channel master setting register	√	×
$013C_H$	DM17CM	DMAC1 channel 7 channel master setting register	√	×
0200_H + 4 × [DTS channel number] *1 (0200_H - $03FC_H$)	DTSnnnCM *1	DTS channel nnn channel master setting register *1	√	×

Note 1. [DTS channel number] and “nnn” in the register symbols and meanings are numbers in the range from 000 to 127.

7.9.2 Details of Global Registers

7.9.2.1 DMACTL — DMA Control Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 8000$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DMA SPD
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 7.11 DMACTL Register Contents

Bit Position	Bit Name	Function
31 to 1	—	Reserved These bits are always read as 0. The write value should be 0.
0	DMASPD	DMA suspension This bit shows whether DMA transfer for all channels is suspended. If a user writes 1 to this bit, DMA transfer for all channels can be suspended. If a user writes 0 to this bit, suspension of DMA transfer for all channels can be cleared. The suspension controlled by this bit is independent from the suspension controlled by the transfer enable bit (DTE) of each DMAC channel and the suspension setting bit (DTSUST) for a DTS. That means, if this bit is 1, all DMA transfers are suspended regardless of the values of the DTE bit of each channel and the DTSUST bit of the DTS. Writing to this bit does not affect the DTE bit of each channel and the DTSUST bit of the DTS. 0: DMA suspension cleared 1: DMA suspension request/DMA suspension ongoing

CAUTION

To suspend or resume currently ongoing DMA transfer by the DTS, the DMA transfer request enable bit (DTFSLnnn.REQEN) must be switched to disable and enable arbitration for the given channel. For details, refer to the notes in **Section 7.3.3, Suspending, Resuming, and Aborting Transfer by the DTS**.

7.9.2.2 DTSCTL1 — DTS Control Register 1

Access: This register can be read/written in 32-bit units.

Address: FFFF 8010_H

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DTSUST
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 7.12 DTSCTL1 Register Contents

Bit Position	Bit Name	Function
31 to 1	—	Reserved These bits are always read as 0. The write value should be 0.
0	DTSUST	DTS suspension This bit shows whether DMA transfer by the DTS is suspended. If a user writes 1 to this bit, DMA transfer by the DTS can be suspended. 0: DTS suspension cleared 1: DTS suspension request/DTS suspension ongoing

CAUTION

To suspend or resume currently ongoing DMA transfer by the DTS, the DMA transfer request enable bit (DTFSLnnn.REQEN) must be switched to disable and enable arbitration for the given channel. For details, refer to the notes in **Section 7.3.3, Suspending, Resuming, and Aborting Transfer by the DTS**.

7.9.2.3 DTSCTL2 — DTS Control Register 2

Access: This register can be read/written in 32-bit units.

Address: FFFF 8014_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DTSTIT
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 7.13 DTSCTL2 Register Contents

Bit Position	Bit Name	Function
31 to 1	—	Reserved These bits are always read as 0. The write value should be 0.
0	DTSTIT	DTS transfer abort request While the DTS is suspended, a user can write 1 to this bit to abort the suspended DMA transfer. When the suspended DMA transfer by the DTS is aborted, the DTSSTS.DTSACT bit is cleared to 0. This bit is always read as 0.

7.9.2.4 DTSSTS — DTS Status Register

Access: This register can be read in 32-bit units.

Address: FFFF 8018$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	DTS CYC	DTSACH[6:0]							DTS ACT
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 7.14 DTSSTS Register Contents

Bit Position	Bit Name	Function
31 to 9	—	Reserved
8	DTSCYC	DMA cycle execution state This bit shows whether a DMA cycle is ongoing in the DTS. 0: DMA cycle is not ongoing. 1: DMA cycle is ongoing.
7 to 1	DTSACH[6:0]	DTS transfer channel If there is a channel in the DTS executing DMA transfer, the channel number is shown. If there is no channel in the DTS executing DMA transfer, the channel number of the last DMA transfer is shown.
0	DTSACT	DTS transfer status This bit shows whether there is a channel in the DTS executing DMA transfer. 0: There is a channel in the DTS executing DMA transfer. 1: There is no channel in the DTS executing DMA transfer. If the DTS is put into the suspended state while there is a channel executing DMA transfer, this bit remains 1. If a DTS transfer abort request is made using the DTSCTL2.DTSTIT bit, the suspended DTS transfer is aborted, and this bit is cleared to 0. When DMA transfer error occurs and the DMA transfer is aborted, this bit is cleared.

7.9.2.5 DMACER — DMAC Error Register

Access: This register can be read in 32-bit units.

Address: FFFF 8020$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DM1ER[7:0]								DM0ER[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 7.15 DMACER Register Contents

Bit Position	Bit Name	Function
31 to 16	—	Reserved
15 to 8	DM1ER[7:0]	DMAC1 DMA transfer error status These bits show the DMA transfer error status of channels 0 through 7 of the DMAC1. Each bit is mapped from the DCSTn.ER bit of each channel of the DMAC1 and is read-only. 0: DMA transfer error is not generated 1: DMA transfer error is generated
7 to 0	DM0ER[7:0]	DMAC0 DMA transfer error status These bits show the DMA transfer error status of channels 0 through 7 of the DMAC0. Each bit is mapped from the DCSTn.ER bit of each channel of the DMAC0 and is read-only. 0: DMA transfer error is not generated 1: DMA transfer error is generated

7.9.2.6 DTSER1 — DTS Error Register 1

Access: This register can be read in 32-bit units.

Address: FFFF 8024$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	DTSERCH[6:0]							—	—	—	—	—	—	DTSERWR	DTSER
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 7.16 DTSER1 Register Contents

Bit Position	Bit Name	Function
31 to 15	—	Reserved
14 to 8	DTSERCH[6:0]	DTS error channel These bits show the DTS channel number of the first DMA transfer error after the DTSER bit is cleared to 0. These bits are read-only and cannot be cleared.
7 to 2	—	Reserved
1	DTSERWR	DTS DMA transfer error occurring cycle This bit is updated at the same time as setting of the DTS DMA transfer error flag (DTSER), indicating which cycle of read or write the DMA transfer error occurs in. This bit is not updated when a new DMA transfer error occurs after the DTSER bit has been set. If the DTSER bit is cleared, this bit is also cleared. 0: DMA transfer error occurs in read cycle. 1: DMA transfer error occurs in write cycle.
0	DTSER	DTS DMA transfer error flag This bit shows whether DMA transfer error is generated in the DTS. 0: DMA transfer error is not generated 1: DMA transfer error is generated If DMA transfer error is generated in the DTS while this bit is 0, this bit is set, and DTSERCH6-0 retains the DTS channel number of the DMA transfer error. If DMA transfer error is generated in the DTS while this bit is 1, this bit remains 1, and DTSERCH6 to 0 does not change. This bit can be cleared by using the DTSERC register.

7.9.2.7 DTSER2 — DTS Error Register 2

Access: This register can be read in 32-bit units.

Address: FFFF 8028$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RMADED	RAMDEDOV	—	—	RAMDEDAD[11:0]											
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RAMSED	RAMSEDOV	—	—	RAMSEDAD[11:0]											
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 7.17 DTSER2 Register Contents (1/2)

Bit Position	Bit Name	Function
31	RAMDED	DTSRAM 2-bit error flag This bit shows whether the 2-bit error is generated in the read access to the DTSRAM. 0: 2-bit error is not generated in the DTSRAM 1: 2-bit error is generated in the DTSRAM If 2-bit error is generated in the DTSRAM with the read access while this bit is 0, this bit is set, and RAMDEDAD11-0 retains the DTSRAM address of the error. If 2-bit error is generated in the DTSRAM with the read access while this bit is 1, this bit remains 1, and RAMDEDAD11-0 do not change. This bit can be cleared by using the DTSERC register.
30	RAMDEDOV	DTSRAM 2-bit error overflow flag This bit is set when the RAMDED bit is 1 and the 2-bit error occurs in DTSRAM read access whose address is different from that specified by the RAMDEDAD11 to 0 bits. This bit can be cleared by operation of the DTSERC register.
29, 28	—	Reserved
27 to 16	RAMDEDAD [11:0]	DTSRAM 2-bit error address These bits show the DTSRAM address of the first DTSRAM 2-bit error with the read access after the RAMDED bit is cleared to 0. These bits are read-only and cannot be cleared.
15	RAMSED	DTSRAM 1-bit error flag This bit shows whether the 1-bit error is generated in the read access to the DTSRAM. 0: 1-bit error is not generated in the DTSRAM 1: 1-bit error is generated in the DTSRAM If 1-bit error is generated in the DTSRAM with the read access while this bit is 0, this bit is set, and RAMSEDAD11 to 0 retains the DTSRAM address of the error. If 1-bit error is generated in the DTSRAM with the read access while this bit is 1, this bit remains 1, and RAMSEDAD11-0 do not change. This bit can be cleared by using the DTSERC register.
14	RAMSEDOV	DTSRAM 1-bit error overflow flag This bit is set when the RAMSED bit is 1 and the 1-bit error occurs in DTSRAM read access whose address is different from that specified by the RAMSEDAD11 to 0 bits. This bit can be cleared by operation of the DTSERC register.
13, 12	—	Reserved

Table 7.17 DTSER2 Register Contents (2/2)

Bit Position	Bit Name	Function
11 to 0	RAMSEDAD [11:0]	DTSRAM 1-bit error address These bits show the DTSRAM address of the first DTSRAM 1-bit error with the read access after the RAMSED bit is cleared to 0. These bits are read-only and cannot be cleared.

7.9.2.8 DTSERC — DTS Error Clear Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 802C$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RAMDEDC	RAMDEDOVC	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RAMSEDC	RAMSEDOVC	—	—	—	—	—	—	—	—	—	—	—	—	—	DTSERC
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 7.18 DTSERC Register Contents

Bit Position	Bit Name	Function
31	RAMDEDC	DTSRAM 2-bit error flag clear If a user writes 1 to this bit, the DTSRAM 2-bit error flag (DTSER2.RAMDED) is cleared. 0 is always read from this bit.
30	RAMDEDOVC	DTSRAM 2-bit error overflow flag clear When the user writes 1 to this bit, the DTSRAM 2-bit error overflow flag (DTSER2.RAMDEDOV) is cleared. The read value of this bit is always 0.
29 to 16	—	Reserved These bits are always read as 0. The write value should be 0.
15	RAMSEDC	DTSRAM 1-bit error flag clear If a user writes 1 to this bit, the DTSRAM 1-bit error flag (DTSER2.RAMSED) is cleared. 0 is always read from this bit.
14	RAMSEDOVC	DTSRAM 1-bit error overflow flag clear When the user writes 1 to this bit, the DTSRAM 1-bit error overflow flag (DTSER2.RAMSEDOV) is cleared. The read value of this bit is always 0.
13 to 1	—	Reserved These bits are always read as 0. The write value should be 0.
0	DTSERC	DTS error flag clear If a user writes 1 to this bit, the DTS DMA error flag (DTSER1.DTSER) is cleared. 0 is always read from this bit.

7.9.2.9 DM0CMV — DMAC0 Register Access Protection Violation Register

Access: This register can be read in 32-bit units.

Address: FFFF 8030_H

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	MINF[6:1]						—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	VCH[2:0]			—	—	—	VF
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 7.19 DM0CMV Register Contents

Bit Position	Bit Name	Function
31 to 23	—	Reserved
22 to 17	MINF[6:1]	Illegal access master information These bits show the accessing master information of the first illegal access after the VF bit is cleared to 0. If illegal access occurs while the VF bit is 1, these bits do not change. These bits are read-only and cannot be cleared. The following master information of the originator of the access is retained in MINF6 to MINF1. MINF6 to 4: The PEID of the originator of the access MINF3, 2: The SPID of the originator of the access MINF1: The UM of the originator of the access
16 to 7	—	Reserved
6 to 4	VCH[2:0]	Illegal access channel These bits show the channel number (0 to 7) of the first illegal access after the VF bit is cleared to 0. If illegal access occurs while the VF bit is 1, these bits do not change. These bits are read-only and cannot be cleared.
3 to 1	—	Reserved
0	VF	Illegal access flag This bit shows whether illegal access occurs in the DMAC0. 0: No illegal access has occurred in the DMAC0 1: Illegal access has occurred in the DMAC0 If illegal access occurs in the DMAC0 while this bit is 0, this bit is set, and MINF6 to 1 and VCH2 to 0 store their respective information. If illegal access occurs in the DMAC0 while this bit is 1, this bit remains 1, and MINF6 to 1 and VCH2 to 0 do not change. This bit can be cleared by using the CMVC register.

7.9.2.10 DM1CMV — DMAC1 Register Access Protection Violation Register

Access: This register can be read in 32-bit units.

Address: FFFF 8034_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	MINF[6:1]						—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	VCH[2:0]			—	—	—	VF
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 7.20 DM1CMV Register Contents

Bit Position	Bit Name	Function
31 to 23	—	Reserved
22 to 17	MINF[6:1]	Illegal access master information These bits show the accessing master information of the first illegal access after the VF bit is cleared to 0. If illegal access occurs while the VF bit is 1, these bits do not change. These bits are read-only and cannot be cleared. The following master information of the originator of the access is retained in MINF6 to MINF1. MINF6 to 4: The PEID of the originator of the access MINF3, 2: The SPID of the originator of the access MINF1: The UM of the originator of the access
16 to 7	—	Reserved
6 to 4	VCH[2:0]	Illegal access channel These bits show the channel number (0 to 7) of the first illegal access after the VF bit is cleared to 0. If illegal access occurs while the VF bit is 1, these bits do not change. These bits are read-only and cannot be cleared.
3 to 1	—	Reserved
0	VF	Illegal access flag This bit shows whether illegal access occurs in the DMAC1. 0: No illegal access has occurred in the DMAC1 1: Illegal access has occurred in the DMAC1 If illegal access occurs in the DMAC1 while this bit is 0, this bit is set, and MINF6 to 1 and VCH2 to 0 store their respective information. If illegal access occurs in the DMAC1 while this bit is 1, this bit remains 1, and MINF6 to 1 and VCH2 to 0 do not change. This bit can be cleared by using the CMVC register.

7.9.2.11 DTSCMV — DTS Register Access Protection Violation Register

Access: This register can be read in 32-bit units.

Address: FFFF 8038_H

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	MINF[6:1]						—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	VCH[6:0]							—	—	—	VF
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 7.21 DTSCMV Register Contents

Bit Position	Bit Name	Function
31 to 23	—	Reserved
22 to 17	MINF[6:1]	Illegal access master information These bits show the accessing master information of the first illegal access after the VF bit is cleared to 0. If illegal access occurs while the VF bit is 1, these bits do not change. These bits are read-only and cannot be cleared. The following master information of the originator of the access is retained in MINF6 to MINF 1. MINF6 to 4: The PEID of the originator of the access MINF3, 2: The SPID of the originator of the access MINF1: The UM of the originator of the access
16 to 11	—	Reserved
10 to 4	VCH[6:0]	Illegal access channel These bits show the channel number (0 to 127) of the first illegal access after the VF bit is cleared to 0. If illegal access occurs while the VF bit is 1, these bits do not change. These bits are read-only and cannot be cleared.
3 to 1	—	Reserved
0	VF	Illegal access flag This bit shows whether illegal access occurs in the DTS. 0: No illegal access has occurred in the DTS 1: Illegal access has occurred in the DTS If illegal access occurs in the DTS while this bit is 0, this bit is set, and MINF6 to 1 and VCH2 to 0 store their respective information. If illegal access occurs in the DTS while this bit is 1, this bit remains 1, and MINF6 to 1 and VCH2 to 0 do not change. This bit can be cleared by using the CMVC register.

7.9.2.12 CMVC — Register Access Protection Violation Clear Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 803C$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	DTSVC	DM1VC	DM0VC
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Table 7.22 CMVC Register Contents

Bit Position	Bit Name	Function
31 to 3	—	Reserved These bits are always read as 0. The write value should be 0.
2	DTSVC	DTS illegal access flag clear The DTS illegal access flag (DTSCMV.VF) can be cleared by writing 1 to this bit. 0 is always read from this bit.
1	DM1VC	DMAC1 illegal access flag clear The DMAC1 illegal access flag (DM1CMV.VF) can be cleared by writing 1 to this bit. 0 is always read from this bit.
0	DM0VC	DMAC0 illegal access flag clear The DMAC0 illegal access flag (DM0CMV.VF) can be cleared by writing 1 to this bit. 0 is always read from this bit.

7.9.2.13 DTSPRn — DTS Channel Priority Setting Register (n = 0 to 7)

- DTSPR0

Access: This register can be read/written in 32-bit units.

Address: FFFF 8060$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DTS15PR[1:0]		DTS14PR[1:0]		DTS13PR[1:0]		DTS12PR[1:0]		DTS11PR[1:0]		DTS10PR[1:0]		DTS9PR[1:0]		DTS8PR[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DTS7PR[1:0]		DTS6PR[1:0]		DTS5PR[1:0]		DTS4PR[1:0]		DTS3PR[1:0]		DTS2PR[1:0]		DTS1PR[1:0]		DTS0PR[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.23 DTSPR0 Register Contents

Bit Position	Bit Name	Function
31 to 0	DTS[15:0] PR[1:0]	DTS channel [15:0] priority setting These bits configure the priority level of each DTS channel used for DTS channel arbitration. 00 is the highest priority, and 11 is the lowest.

- DTSPR1

Access: This register can be read/written in 32-bit units.

Address: FFFF 8064$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DTS31PR[1:0]		DTS30PR[1:0]		DTS29PR[1:0]		DTS28PR[1:0]		DTS27PR[1:0]		DTS26PR[1:0]		DTS25PR[1:0]		DTS24PR[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DTS23PR[1:0]		DTS22PR[1:0]		DTS21PR[1:0]		DTS20PR[1:0]		DTS19PR[1:0]		DTS18PR[1:0]		DTS17PR[1:0]		DTS16PR[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.24 DTSPR1 Register Contents

Bit Position	Bit Name	Function
31 to 0	DTS[31:16] PR[1:0]	DTS channel [31:16] priority setting These bits configure the priority level of each DTS channel used for DTS channel arbitration. 00 is the highest priority, and 11 is the lowest.

- DTSPR2

Access: This register can be read/written in 32-bit units.

Address: FFFF 8068$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DTS47PR[1:0]		DTS46PR[1:0]		DTS45PR[1:0]		DTS44PR[1:0]		DTS43PR[1:0]		DTS42PR[1:0]		DTS41PR[1:0]		DTS40PR[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DTS39PR[1:0]		DTS38PR[1:0]		DTS37PR[1:0]		DTS36PR[1:0]		DTS35PR[1:0]		DTS34PR[1:0]		DTS33PR[1:0]		DTS32PR[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.25 DTSPR2 Register Contents

Bit Position	Bit Name	Function
31 to 0	DTS[47:32] PR[1:0]	DTS channel [47:32] priority setting These bits configure the priority level of each DTS channel used for DTS channel arbitration. 00 is the highest priority, and 11 is the lowest.

- DTSPR3

Access: This register can be read/written in 32-bit units.

Address: FFFF 806C$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DTS63PR[1:0]		DTS62PR[1:0]		DTS61PR[1:0]		DTS60PR[1:0]		DTS59PR[1:0]		DTS58PR[1:0]		DTS57PR[1:0]		DTS56PR[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DTS55PR[1:0]		DTS54PR[1:0]		DTS53PR[1:0]		DTS52PR[1:0]		DTS51PR[1:0]		DTS50PR[1:0]		DTS49PR[1:0]		DTS48PR[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.26 DTSPR3 Register Contents

Bit Position	Bit Name	Function
31 to 0	DTS[63:48] PR[1:0]	DTS channel [63:48] priority setting These bits configure the priority level of each DTS channel used for DTS channel arbitration. 00 is the highest priority, and 11 is the lowest.

- DTSPR4

Access: This register can be read/written in 32-bit units.

Address: FFFF 8070$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DTS79PR[1:0]		DTS78PR[1:0]		DTS77PR[1:0]		DTS76PR[1:0]		DTS75PR[1:0]		DTS74PR[1:0]		DTS73PR[1:0]		DTS72PR[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DTS71PR[1:0]		DTS70PR[1:0]		DTS69PR[1:0]		DTS68PR[1:0]		DTS67PR[1:0]		DTS66PR[1:0]		DTS65PR[1:0]		DTS64PR[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.27 DTSPR4 Register Contents

Bit Position	Bit Name	Function
31 to 0	DTS[79:64] PR[1:0]	DTS channel [79:64] priority setting These bits configure the priority level of each DTS channel used for DTS channel arbitration. 00 is the highest priority, and 11 is the lowest.

- DTSPR5

Access: This register can be read/written in 32-bit units.

Address: FFFF 8074$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DTS95PR[1:0]		DTS94PR[1:0]		DTS93PR[1:0]		DTS92PR[1:0]		DTS91PR[1:0]		DTS90PR[1:0]		DTS89PR[1:0]		DTS88PR[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DTS87PR[1:0]		DTS86PR[1:0]		DTS85PR[1:0]		DTS84PR[1:0]		DTS83PR[1:0]		DTS82PR[1:0]		DTS81PR[1:0]		DTS80PR[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.28 DTSPR5 Register Contents

Bit Position	Bit Name	Function
31 to 0	DTS[95:80] PR[1:0]	DTS channel [95:80] priority setting These bits configure the priority level of each DTS channel used for DTS channel arbitration. 00 is the highest priority, and 11 is the lowest.

- DTSPR6

Access: This register can be read/written in 32-bit units.

Address: FFFF 8078$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DTS111PR[1:0]		DTS110PR[1:0]		DTS109PR[1:0]		DTS108PR[1:0]		DTS107PR[1:0]		DTS106PR[1:0]		DTS105PR[1:0]		DTS104PR[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DTS103PR[1:0]		DTS102PR[1:0]		DTS101PR[1:0]		DTS100PR[1:0]		DTS99PR[1:0]		DTS98PR[1:0]		DTS97PR[1:0]		DTS96PR[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.29 DTSPR6 Register Contents

Bit Position	Bit Name	Function
31 to 0	DTS[111:96] PR[1:0]	DTS channel [111:96] priority setting These bits configure the priority level of each DTS channel used for DTS channel arbitration. 00 is the highest priority, and 11 is the lowest.

- DTSPR7

Access: This register can be read/written in 32-bit units.

Address: FFFF 807C$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DTS127PR[1:0]		DTS126PR[1:0]		DTS125PR[1:0]		DTS124PR[1:0]		DTS123PR[1:0]		DTS122PR[1:0]		DTS121PR[1:0]		DTS120PR[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DTS119PR[1:0]		DTS118PR[1:0]		DTS117PR[1:0]		DTS116PR[1:0]		DTS115PR[1:0]		DTS114PR[1:0]		DTS113PR[1:0]		DTS112PR[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.30 DTSPR7 Register Contents

Bit Position	Bit Name	Function
31 to 0	DTS[127:112] PR[1:0]	DTS channel [127:112] priority setting These bits configure the priority level of each DTS channel used for DTS channel arbitration. 00 is the highest priority, and 11 is the lowest.

7.9.2.14 DTRECCTL — DTSRAM ECC Control Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 8080$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PROT[1:0]		—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SEC DIS	ECC DIS
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Table 7.31 DTRECCTL Register Contents

Bit Position	Bit Name	Function
31, 30	PROT[1:0]	These bits enable or disable writing to the ECCDIS and SECDIS bits. The written data is not retained. The read value is always 0. These bits should be written when (PROT1, PROT0) = (0, 1)
29 to 2	—	Reserved These bits are always read as 0. The write value should be 0.
1	SECDIS	DTSRAM SEC error correction disable This bit enables or disables SEC error correction when the ECCDIS bit is 0. 1-bit ECC error detection operation is always executed when ECCDIS bit is 0 regardless of the state of this bit. Set (PROT1, PROT0) = (0, 1) at the same time as writing to this bit. 0: Error correction is enabled when the SEC error is detected. 1: Error correction is disabled when the SEC error is detected.
0	ECCDIS	DTSRAM ECC disable This bit enables or disables DTSRAM ECC error detection and correction. Set (PROT1, PROT0) = (0, 1) at the same time as writing to this bit. 0: Error detection and correction are enabled. 1: Error detection and correction are disabled. The encoding function is effective when error detection and correction are disabled

7.9.2.15 DTRERINT — DTSRAM Error Notification Control Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 8084_H

Value after reset: 0000 0002_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DEDIE	SEDIE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Table 7.32 DTRERINT Register Contents

Bit Position	Bit Name	Function
31 to 2	—	Reserved These bits are always read as 0. The write value should be 0.
1	DEDIE	DTSRAM 2-bit error notification enable This bit enables or disables notification of 2-bit error detection to the ECM when the ECCDIS bit in DTRECCTL is 0. 0: Notification of 2-bit error to ECM is disabled. 1: Notification of 2-bit error to ECM is enabled.
0	SEDIE	DTSRAM 1-bit error notification enable This bit enables or disables notification of 1-bit error detection to the ECM when the ECCDIS bit in DTRECCTL is 0. 0: Notification of 1-bit error to ECM is disabled. 1: Notification of 1-bit error to ECM is enabled.

7.9.2.16 DTRTSCTL — DTSRAM Test Control Register

This register is used for ECC test (self-diagnosis). It enables setting of ECC test mode and selection of ECC data to be written to the DTSRAM.

Access: This register can be read/written in 32-bit units.

Address: FFFF 8094$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PROT[1:0]		—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ECC TST	DAT SEL
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Table 7.33 DTRTSCTL Register Contents

Bit Position	Bit Name	Function
31, 30	PROT[1:0]	These bits enable or disable writing to the ECCTST and DATSEL bits. The written data is not retained. The read value is always 0. These bits should be written when (PROT1, PROT0) = (0, 1)
29 to 2	—	Reserved These bits are always read as 0. The write value should be 0.
1	ECCTST	DTSRAM ECC Test Mode This bit enables or disables DTSRAM ECC test mode. Set (PROT1, PROT0) = (0, 1) at the same time as writing to this bit. 0: ECC test mode is enabled. 1: ECC test mode is disabled.
0	DATSEL	ECC Test Data Selection This bit is valid when ECCTST is 1 and selects ECC data to be written to the DTSRAM. Set (PROT1, PROT0) = (0, 1) at the same time as writing to this bit. 0: ECC encoded from the written data is used. 1: The value specified by the DTSRAM test write data register (DTRTWDAT) is used.

7.9.2.17 DTRTWDAT — DTSRAM Test Write Data Register

This register is used for ECC test (self-diagnosis). It specifies ECC data to be written to the DTSRAM after ECC test mode is enabled (ECCTST = 1).

Access: This register can be read/written in 32-bit units.

Address: FFFF 8098$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	TWDAT[6:0]						
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.34 DTRTWDAT Register Contents

Bit Position	Bit Name	Function
31 to 7	—	Reserved These bits are always read as 0. The write value should be 0.
6 to 0	TWDAT[6:0]	ECC Test Write Data These bits specify ECC data to be written to the DTSRAM when DTRTSCTL.ECCTST = 1 and DTRTSCTL.DATSEL = 1. Writing to these bits is enabled when DTRTSCTL.ECCTST = 1 When DTRTSCTL.ECCTST = 0, these bits cannot be written and its read value is 0.

7.9.2.18 DTRTRDAT — DTSRAM Test Read Data Register

This register is used for ECC test (self-diagnosis). It reads out ECC data of the DTSRAM after ECC test mode is enabled (ECCTST = 1).

Access: This register can be read in 32-bit units.

Address: FFFF 809C$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	TRDAT[6:0]						
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 7.35 DTRTRDAT Register Contents

Bit Position	Bit Name	Function
31 to 7	—	Reserved
6 to 0	TRDAT[6:0]	ECC Test Read Data These bits retain the last ECC data read out from the DTSRAM when DTRTSCTL.ECCTST = 1. When DTRTSCTL.ECCTST = 0, the read value of these bits is 0.

7.9.2.19 DMnnCM — DMAC Channel Master Setting Register (nn = 00 to 07 and 10 to 17)

Access: This register can be read/written in 32-bit units.

Address: FFFF 8100_H + 4_H × Ch. No. n (n = 0 to 7)
FFFF 8120_H + 4_H × Ch. No. n - 10 (n = 10 to 17)

Value after reset: 0000 0010_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	PEID[2:0]			SPID[1:0]		UM	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R

Table 7.36 DMnnCM Register Contents

Bit Position	Bit Name	Function ion
31 to 7	—	Reserved These bits are always read as 0. The write value should be 0.
6 to 4	PEID[2:0]	Channel master PEID setting Specify the PEID information of the master assigned to the channel.
3, 2	SPID[1:0]	Channel master SPID setting Specify the SPID information used by the master assigned to the channel.
1	UM	Channel master UM setting Specifies the UM information of the master assigned to the channel.
0	—	Reserved This bit is always read as 0. The write value should be 0.

CAUTION

DM00CM to DM07CM configure the channel master information of the DMAC0 channel 0 to 7 respectively.

DM10CM to DM17CM configure the channel master information of the DMAC1 channel 0 to 7 respectively.

For information about the functions this register offers, see **Section 7.5, Reliability Function**.

7.9.2.20 DTSnnnCM — DTS Channel Master Setting Register (nnn = 000 to 127)

Access: This register can be read/written in 32-bit units.

Address: FFFF 8200_H + 4_H × Ch. No. n (n = 0 to 127)

Value after reset: Undefined

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	PEID[2:0]			SPID[1:0]		UM	—
Value after reset	Undefined															
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CMC[15:0]															
Value after reset	Undefined															
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.37 DTS Channel Master Setting Register Contents

Bit Position	Bit Name	Function
31 to 23	—	Reserved When read, the value returned is undefined. The write value should be 0.
22 to 20	PEID[2:0]	Channel master PEID setting Specify the PEID information of the master assigned to the channel.
19, 18	SPID[1:0]	Channel master SPID setting Specify the SPID information used by the master assigned to the channel.
17	UM	Channel master UM setting Specifies the UM information of the master assigned to the channel.
16	—	Reserved When read, the value returned is undefined. The write value should be 0.
15 to 0	CMC[15:0]	Transfer count compare In terms of contents, this field is the same as bits [15:0] of **Section 7.11.3.8, DTTCCnnn — DTS Transfer Count Compare Register**.

CAUTION

DTS000CM to DTS127CM configure the channel master information of the DTS channel 0 to 127 respectively.

For information about the functions this register offers, see **Section 7.5, Reliability Function**.

CAUTION

The 16 low-order bits of this register are shared with the DTS transfer count compare register, one of the DTS channel registers.

If you write to this register, the DTS transfer count compare register is updated as well.

Recommended setup procedure of the DTS channel master setting register

When the special master configures the overall DMA operation, the channel master setting must be configured in bits 22 to 17 in this register, and bits 15 to 0 and the reserved bits (bits 31 to 23 and 16) must be initialized with 0.

When the master assigned to the channel updates the value of the transfer count compare, use the DTS transfer count compare register instead.

Bits 31 to 23 of this register are reserved, but you can read and write those bits. Our recommendation is as follows. When you write to those bits, write 0. When you read from those bits, discard the read value by software.

7.10 DMAC Channel Register

7.10.1 DMAC Channel Register Address

Address = Base address "FFFF 8000_H" + Offset address

Table 7.38 DMAC Channel Register Address

Offset Address	Register Symbol	Meaning	Accessed Permission	
			Special Master	General Master
$0400_H + 40_H \times$ [channel number]	DSAn	DMAC source address	√	√
$0404_H + 40_H \times$ [channel number]	DDAn	DMAC destination address	√	√
$0408_H + 40_H \times$ [channel number]	DTCn	DMAC transfer count	√	√
$040C_H + 40_H \times$ [channel number]	DTCTn	DMAC transfer control	√	√
$0410_H + 40_H \times$ [channel number]	DRSAn	DMAC reload source address	√	√
$0414_H + 40_H \times$ [channel number]	DRDAn	DMAC reload destination address	√	√
$0418_H + 40_H \times$ [channel number]	DRTCn	DMAC reload transfer count	√	√
$041C_H + 40_H \times$ [channel number]	DTCCn	DMAC transfer count compare	√	√
$0420_H + 40_H \times$ [channel number]	DCENn	DMAC channel operation enable setting	√	√
$0424_H + 40_H \times$ [channel number]	DCSTn	DMAC transfer status	√	√
$0428_H + 40_H \times$ [channel number]	DCSTSn	DMAC transfer status set	√	√
$042C_H + 40_H \times$ [channel number]	DCSTCn	DMAC transfer status clear	√	√
$0430_H + 40_H \times$ [channel number]	DTFRn	DTFR setting	√	√
$0434_H + 40_H \times$ [channel number]	DTFRRQn	DTFR transfer request status	√	√
$0438_H + 40_H \times$ [channel number]	DTFRRQCn	DTFR transfer request clear	√	√

Note 1. The [channel number] in the offset addresses and "n" in the register symbols are numbers in the range from 0 to 15, and the correspondence between the channel number n and the channel is as follows.

Channel number n	Channel
0	DMAC0 channel 0
1	DMAC0 channel 1
2	DMAC0 channel 2
3	DMAC0 channel 3
4	DMAC0 channel 4
5	DMAC0 channel 5
6	DMAC0 channel 6
7	DMAC0 channel 7
8	DMAC1 channel 0
9	DMAC1 channel 1
10	DMAC1 channel 2
11	DMAC1 channel 3
12	DMAC1 channel 4
13	DMAC1 channel 5
14	DMAC1 channel 6
15	DMAC1 channel 7

7.10.2 Details of DMAC Channel Registers

The “n” in the register symbols indicates the DMA channel number (n = 0 to 15).

7.10.2.1 DSAn — DMAC Source Address Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 8400_H + 40_H × Ch. No. n (n = 0 to 15)

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SA[31:16]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SA[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.39 DSAn Register Contents

Bit Position	Bit Name	Function
31 to 0	SA[31:0]	Source address Specifies the DMA transfer source address. Those bits are updated whenever a DMA cycle is executed. If you read from those bits, the transfer source address for the next DMA cycle is read.

CAUTIONS

1. **It is forbidden to write to those bits when the channel operation is enabled (DTE bit = 1). If you do, the correct operation is not guaranteed.**
2. **The address must be set up while the DTE bit is 0.**
3. **DMA transfer for misaligned data is not supported. The possible lowest four bits of the address for each transfer data size is as follows. (× denotes an arbitrary one bit.)**
   The correct operation is not guaranteed if you select other setting.

Data Size	SA3	SA2	SA1	SA0
8 bits	×	×	×	×
16 bits	×	×	×	0
32 bits	×	×	0	0
64 bits	×	0	0	0
128 bits	0	0	0	0

7.10.2.2 DDAn — DMAC Destination Address Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 8404_H + 40_H × Ch. No. n (n = 0 to 15)

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DA[31:16]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DA[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.40 DDAn Register Contents

Bit Position	Bit Name	Function
31 to 0	DA[31:0]	Destination address Specifies the DMA transfer destination address. Those bits are updated whenever a DMA cycle is executed. If you read from those bits, the transfer destination address for the next DMA cycle is read.

CAUTIONS

1. **It is forbidden to write to those bits when the channel operation is enabled (DTE bit = 1). If you do, the correct operation is not guaranteed.**
2. **The address must be set up while the DTE bit is 0.**
3. **If transfer error is generated in the read cycle of DMA transfer, the write cycle is not executed, but the destination address is updated.**
4. **DMA transfer for misaligned data is not supported. The possible lowest four bits of the address for each transfer data size is as follows. (× denotes an arbitrary one bit.)**
   The correct operation is not guaranteed if you select other setting.

Data Size	DA3	DA2	DA1	DA0
8 bits	×	×	×	×
16 bits	×	×	×	0
32 bits	×	×	0	0
64 bits	×	0	0	0
128 bits	0	0	0	0

7.10.2.3 DTCn — DMAC Transfer Count Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 8408_H + 40_H × Ch. No. n (n = 0 to 15)

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ARC[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TRC[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.41 DTCn Register Contents

Bit Position	Bit Name	Function
31 to 16	ARC[15:0]	Address reload count Specifies the number of transfers until the address reload when the reload function 2 is used, and also specifies the number of transfers when the block transfer 2 is used. If you read from those bits during DMA transfer, the address reload count for the next DMA cycle is read. When the reload function 2 or block transfer 2 is used, ARC[15:0] bits are decremented by one for every DMA cycle. When the reload function 2 or block transfer 2 is not used, ARC[15:0] bits are not updated. If the value is 0000_H, it means that the number of transfers until the address reload when the reload function 2 is used and the number of transfers when the block transfer 2 is used are 65536.
15 to 0	TRC[15:0]	Transfer count Configures the number of transfers. TRC[15:0] bits are decremented by one whenever a DMA cycle is executed. If you read from those bits, the remaining number of transfers for the next DMA cycle is read. If the reload function is not used, after the last transfer is complete, the value at the completion (0000_H) is retained.

TRC[15:0]	Operation
0000_H	The number of transfers is 65536, or the transfer is complete.
0001_H	The number of transfers or remaining transfers is 1.
:	:
$FFFF_H$	The number of transfers or remaining transfers is 65535.

CAUTIONS

1. **It is forbidden to write to those bits when the channel operation is enabled (DTE bit = 1). If you do, the correct operation is not guaranteed.**
2. **If transfer error is generated in the read cycle of DMA transfer, the write cycle is not executed, but the transfer count and the address reload count are updated.**

7.10.2.4 DTCTn — DMAC Transfer Control Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 840C$_H$+ 40$_H$ × Ch. No. n (n = 0 to 15)

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	ESE	DRS	—	—	—	—	—	CHNSEL[2:0]			CHNE[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CCE	TCE	MLE	RLD2M[1:0]		RLD1M[1:0]		DACM[1:0]		SACM[1:0]		DS[2:0]			TRM[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.42 DTCTn Register Contents (1/3)

Bit Position	Bit Name	Function
31 to 28	—	Reserved These bits are always read as 0. The write value should be 0.
27	ESE	Transfer error case DMA transfer disable setting Configures whether a DMA cycle is executed when the DCSTn.ER bit is set due to DMA transfer error. If this bit is cleared to 0, even when the DCSTn.ER bit is set due to DMA transfer error, the following DMA cycles can be executed. If this bit is set to 1, the following DMA cycles are not executed when the DCSTn.ER bit is set due to DMA transfer error. 0: DMA cycles are executed while the DCSTn.ER bit is set. 1: DMA cycles are not executed while the DCSTn.ER bit is set.
26	DRS	DMA transfer request selection assignment Selects the type of DMA transfer requests to be accepted. 0: Software DMA transfer request 1: Hardware DMA transfer request
25 to 21	—	Reserved These bits are always read as 0. The write value should be 0.
20 to 18	CHNSEL[2:0]	Next channel in the chain Specify the next channel in the chain. The next channel must be another channel in the same DMAC. You cannot specify a channel in the different DMAC or in the DTS. You cannot specify the same channel for the next channel (operation is not guaranteed if the setting is made).
17, 16	CHNE[1:0]	Chain enable Select the chain function. 00: Disabled 01: Chain at the last transfer A chain request is generated at the completion of the DMA cycle in which the remaining transfer count is one. 10: (Setting prohibited. Operation is not guaranteed if the setting is made. 11: Always chain A chain request is generated at the completion of every DMA cycle.
15	CCE	Transfer count match interrupt enable If this bit is set, a transfer count match interrupt is generated at the completion of the DMA cycle in which the transfer count compare register and the remaining transfer count have the same value.
14	TCE	Transfer completion interrupt enable If this bit is set, a transfer completion interrupt is generated at the completion of the last transfer.

Table 7.42 DTCTn Register Contents (2/3)

Bit Position	Bit Name	Function
13	MLE	Continuous transfer enable If this bit is set, the DTE bit is not cleared at the completion of DMA transfer. In addition, even if the TC bit is not cleared, DMA transfer starts when there is a DMA transfer request. 0: The DTE bit is cleared at the completion of DMA transfer. In addition, the next DMA transfer can start only after the TC bit is cleared. 1: The DTE bit is not cleared at the completion of DMA transfer. In addition, even if the TC bit is not cleared, DMA transfer starts when there is a DMA transfer request.
12, 11	RLD2M[1:0]	Reload function 2 setting Configure the reload function 2. 00: Reload function 2 is disabled. 01: Reload function 2 is enabled. The source address and address reload count are reloaded at the completion of the DMA cycle in which the address reload count is 1. 10: Reload function 2 is enabled. The destination address and address reload count are reloaded at the completion of the DMA cycle in which the address reload count is 1. 11: Reload function 2 is enabled. The source address, destination address, and address reload count are reloaded at the completion of the DMA cycle in which the address reload count is 1.
10, 9	RLD1M[1:0]	Reload function 1 setting Configure the reload function 1. 00: Reload function 1 is disabled. 01: Reload function 1 is enabled. The source address and transfer count are reloaded at the completion of the DMA cycle in which the remaining transfer count is 1. (If the reload function 2 is enabled, the address reload count is also reloaded.) 10: Reload function 1 is enabled. The destination address and transfer count are reloaded at the completion of the DMA cycle in which the remaining transfer count is 1. (If the reload function 2 is enabled, the address reload count is also reloaded.) 11: Reload function 1 is enabled. The source address, destination address, and transfer count are reloaded at the completion of the DMA cycle in which the remaining transfer count is 1. (If the reload function 2 is enabled, the address reload count is also reloaded.)
8, 7	DACM[1:0]	Destination address count direction Specify the count direction of the destination address. **DACM1 / DACM0 / Direction of Counting** 0 / 0 / Increment 0 / 1 / Decrement 1 / 0 / Fixed 1 / 1 / Setting prohibited (operation not guaranteed)
6, 5	SACM[1:0]	Source address count direction Specify the count direction of the source address. **SACM1 / SACM0 / Direction of Counting** 0 / 0 / **Increment** 0 / 1 / Decrement 1 / 0 / Fixed 1 / 1 / Setting prohibited (operation not guaranteed)

Table 7.42 DTCTn Register Contents (3/3)

<table>
<tr><th>Bit Position</th><th>Bit Name</th><th>Function</th></tr>
<tr><td>4 to 2</td><td>DS[2:0]</td><td>Transfer data size
Specify the transfer data size.
<table>
<tr><th>DS2</th><th>DS1</th><th>DS0</th><th>Transfer Data Size</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td>8 bits</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>16 bits</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>32 bits</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>64 bits</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>128 bits</td></tr>
<tr><td colspan="3">Other than the above</td><td>Setting prohibited (operation not guaranteed)</td></tr>
</table></td></tr>
<tr><td>1, 0</td><td>TRM[1:0]</td><td>Transfer mode
Specify the DMA transfer mode.
00: Single transfer
01: Block transfer 1 (The number of transfers is specified by the transfer count.)
10: Block transfer 2 (The number of transfers is specified by the address reload count.)
11: Setting prohibited (operation not guaranteed)</td></tr>
</table>

CAUTIONS

1. **While channel operation is enabled (DTE bit = 1), changing the value of the MLE bit from “1” to “0” is only possible (correct operation is not guaranteed when the value of any bit other than the MLE bit is changed).**
2. **Correction operation is not guaranteed when the prohibited setting has been set to a given bit.**

7.10.2.5 DRSAn — DMAC Reload Source Address Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 8410_H + 40_H × Ch. No. n (n = 0 to 15)

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RSA[31:16]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RSA[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.43 DRSAn Register Contents

Bit Position	Bit Name	Function
31 to 0	RSA[31:0]	Reload source address Specifies the source address to be reloaded to the DMA source address register when the reload function 1 or reload function 2 is used.

CAUTION

DMA transfer for misaligned data is not supported. The possible lowest four bits of the address for each transfer data size is as follows. (× denotes an arbitrary one bit.) The correct operation is not guaranteed if you select other setting.

Data Size	RSA3	RSA2	RSA1	RSA0
8 bits	×	×	×	×
16 bits	×	×	×	0
32 bits	×	×	0	0
64 bits	×	0	0	0
128 bits	0	0	0	0

7.10.2.6 DRDAn — DMAC Reload Destination Address Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 8414_H + 40_H × Ch. No. n (n = 0 to 15)

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RDA[31:16]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RDA[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.44 DRDAn Register Contents

Bit Position	Bit Name	Function
31 to 0	RDA[31:0]	Reload destination address Specifies the destination address to be reloaded to the DMA destination address register when the reload function 1 or reload function 2 is used.

CAUTION

DMA transfer for misaligned data is not supported. The possible lowest four bits of the address for each transfer data size is as follows. (× denotes an arbitrary one bit.) The correct operation is not guaranteed if you select other setting.

Data Size	RDA3	RDA2	RDA1	RDA0
8 bits	×	×	×	×
16 bits	×	×	×	0
32 bits	×	×	0	0
64 bits	×	0	0	0
128 bits	0	0	0	0

7.10.2.7 DRTCn — DMAC Reload Transfer Count Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 8418$_H$ + 40$_H$H × Ch. No. n (n = 0 to 15)

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RARC[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RTRC[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.45 DRTCn Register Contents

Bit Position	Bit Name	Function
31 to 16	RARC[15:0]	Reload address reload count Specifies the value to be loaded to the address reload count in the transfer count register at the timing of reload when the reload function 2 is used.
15 to 0	RTRC[15:0]	Reload transfer count Specifies the value to be loaded to the transfer count in the transfer count register at the timing of reload when the reload function 1 is used.

7.10.2.8 DTCCn — DMAC Transfer Count Compare Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 841C$_H$ + 40$_H$ × Ch. No. n (n = 0 to 15)

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CMC[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.46 DTCCn Register Contents

Bit Position	Bit Name	Function
31 to 16	—	Reserved These bits are always read as 0. The write value should be 0.
15 to 0	CMC[15:0]	Transfer count compare Configure the transfer count to be compared to the transfer count register. At the completion of the DMA cycle in which the remaining transfer count is the same as the value in this register, the transfer count match flag (DCSTn.CC) in the DMAC transfer status register is set. Furthermore, if the transfer count match interrupt enable (DTCTn.CCE) bit is 1, a transfer count match interrupt is generated. If 0000$_H$ is set, comparison with the transfer count is disabled. In this case, the transfer count match flag in the DMAC transfer status register is never set, and a transfer count match interrupt is never generated.

CAUTION

It is forbidden to write to those bits when the channel operation is enabled (DTE bit = 1). If you do, the correct operation is not guaranteed.

7.10.2.9 DCENn — DMAC Channel Operation Enable Setting Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 8420_H + 40_H × Ch. No. n (n = 0 to 15)

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DTE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 7.47 DCENn Register Contents

Bit Position	Bit Name	Function
31 to 1	—	Reserved These bits are always read as 0. The write value should be 0.
0	DTE	Channel operation enable Specifies whether to enable or disable the transfer operation of the channel. If the DTE bit is 1, DMA transfer starts when there is a DMA transfer request. If the MLE bit is 0, the DTE bit is cleared automatically at the completion of the DMA transfer. In addition, if 0 is written to the DTE bit during DMA transfer, the DMA transfer is suspended. If 1 is written to the DTE bit during suspension, the suspension is cleared and the DMA transfer resumes. 0: Channel operation is disabled/Channel suspended 1: Channel operation is enabled/Channel suspension cleared

7.10.2.10 DCSTn — DMAC Transfer Status Register

Access: This register can be read in 32-bit units.

Address: FFFF 8424_H + 40_H × Ch. No. n (n = 0 to 15)

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	ERWR	—	—	CY	ER	—	CC	TC	—	—	DR	SR
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 7.48 DCSTn Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 12	—	Reserved
11	ERWR	DMA Transfer Error occurring cycle This bit is updated at the same time as setting of the DMA transfer error flag (ER), indicating which cycle of read or write the DMA transfer error occurs in. This bit is not updated when a new DMA transfer error occurs after the ER bit has been set. If the ER bit is cleared, this bit is also cleared to 0. 0: DMA transfer error occurs in read cycle. 1: DMA transfer error occurs in write cycle.
10, 9	—	Reserved
8	CY	DMA cycle execution state This bit shows whether a DMA cycle is ongoing in this channel. 0: DMA cycle is not ongoing. 1: DMA cycle is ongoing.
7	ER	Transfer error flag This bit is set when DMA transfer error is generated. If this bit is 1 and the DTCTn.ESE bit is set, a DMA cycle is not executed when a DMA transfer request is generated. 0: No DMA transfer error is generated 1: DMA transfer error is generated
6	—	Reserved
5	CC	Transfer count match flag This bit is set at the completion of the DMA cycle in which the remaining transfer count is the same as the value set in the transfer compare register. 0: No compare match has occurred with the transfer count compare register. 1: Compare match has occurred with the transfer count compare register.
4	TC	Transfer completion flag This bit is set at the completion of the last transfer and shows whether the DMA transfer is complete. If the MLE bit is 0 and this bit is 1, a DMA cycle is not executed when a DMA transfer request is generated. 0: DMA transfer incomplete 1: DMA transfer complete
3, 2	—	Reserved

Table 7.48 DCSTn Register Contents (2/2)

Bit Position	Bit Name	Function
1	DR	Hardware DMA transfer request status This bit shows whether there is a hardware DMA transfer request (DMARQ) from the DTFR. This bit changes regardless of the value of the DTE bit when a hardware DMA transfer request from the DTFR is generated. If the software DMA transfer request is selected by the transfer request selection bit (DRS) in the DMAC transfer control register, this bit is not set even when a hardware DMA transfer request is input from the DTFR. 0: There is no hardware DMA transfer request 1: There is a hardware DMA transfer request
0	SR	Software DMA transfer request flag This bit shows whether there is a software DMA transfer request (DMARQ). This bit is automatically cleared at the completion of the last transfer. A user can set this bit by writing 1 to the SRS bit in the DMAC transfer status set register. In addition, a user can clear this bit by writing 1 to the SRC bit in the DMAC transfer status clear register, but if this is done, the ongoing DMA transfer is aborted and cannot be resumed. 0: There is no software DMA transfer request 1: There is a software DMA transfer request

7.10.2.11 DCSTSn — DMAC Transfer Status Set Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 8428_H + 40_H × Ch. No. n (n = 0 to 15)

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SRS
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 7.49 DCSTSn Set Register Contents

Bit Position	Bit Name	Function
31 to 1	—	Reserved These bits are always read as 0. The write value should be 0.
0	SRS	Software DMA transfer request flag A user can set the software DMA transfer request flag (SR) by writing 1 to this bit. 0 is always read from this bit.

7.10.2.12 DCSTCn — DMAC Transfer Status Clear Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 842C$_H$ + 40$_H$ × Ch. No. n (n = 0 to 15)

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	ERC	—	CCC	TCC	—	—	—	SRC
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R	R/W	R/W	R	R	R	R/W

Table 7.50 DCSTCn Register Contents

Bit Position	Bit Name	Function
31 to 8	—	Reserved These bits are always read as 0. The write value should be 0.
7	ERC	Transfer error flag clear The DMA transfer error flag (ER) can be cleared by writing 1 to this bit. 0 is always read from this bit.
6	—	Reserved This bit is always read as 0. The write value should be 0.
5	CCC	Transfer count match flag clear The transfer count match flag (CC) can be cleared by writing 1 to this bit. 0 is always read from this bit.
4	TCC	Transfer completion flag clear The transfer completion flag (TC) can be cleared by writing 1 to this bit. 0 is always read from this bit.
3 to 1	—	Reserved These bits are always read as 0. The write value should be 0.
0	SRC	Software DMA transfer request flag clear The software DMA transfer request flag (SR) can be cleared by writing 1 to this bit. 0 is always read from this bit.

7.10.2.13 DTFRn — DTFR Setting Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 8430_H + 40_H × Ch. No. n (n = 0 to 15)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	REQSEL[6:0]							REQEN
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.51 DTFRn Register Contents

Bit Position	Bit Name	Function
31 to 8	—	Reserved These bits are always read as 0. The write value should be 0.
7 to 1	REQSEL[6:0]	Hardware DMA transfer source selection Select one out of 128 hardware DMA transfer sources as the hardware DMA transfer request. 000_0000: Selecting the DMACTRG[0] input : 111_1111: Selecting the DMACTRG[127] input
0	REQEN	Hardware DMA transfer source selection enable This bit enables/disables the hardware DMA transfer source selection. 0: Hardware DMA transfer source selection is disabled. 1: Hardware DMA transfer source selection is enabled. If this bit is 0, even when the hardware DMA transfer source selected by the REQSEL6-0 bits is activated, it is not recognized as a hardware DMA transfer request, and a hardware DMA transfer request is not generated.

7.10.2.14 DTFRRQn — DTFR Transfer Request Status Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 8434_H + 40_H × Ch. No. n (n = 0 to 15)

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DRQ
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 7.52 DTFRRQn Register Contents

Bit Position	Bit Name	Function
31 to 1	—	Reserved
0	DRQ	Hardware DMA transfer request status If this bit is set, it means that a hardware DMA transfer request exists or is retained. • If the hardware DMA transfer request is an edge detection type*1 This bit shows whether a hardware DMA transfer request generated by edge detection is retained. When the DMA transfer request acceptance signal (DMAAKn) from the DMAC is asserted, this bit is automatically cleared. A user can clear this bit by writing 1 to the DTFRRQCn.DRQC bit. • If the hardware DMA transfer request is a level input type*1 This bit shows whether there is a hardware DMA transfer request input. Even when the DMA transfer request acceptance signal (DMAAKn) from the DMAC is asserted, this bit is not automatically cleared. In addition, this bit is not cleared even when a user writes to the DTFRRQCn.DRQC bit. This bit changes regardless of the value of the DTFRn.REQEN bit when a hardware DMA transfer request is generated. 0: There is no hardware DMA transfer request 1: There is a hardware DMA transfer request

Note 1. Whether the hardware DMA transfer request is an edge detection type or level input type depends on the hardware DMA transfer source selected by DTFRn.REQSEL. The hardware DMA transfer request is only an edge detection type.

7.10.2.15 DTFRRQCn — DTFR Transfer Request Clear Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 8438_H + 40_H × Ch. No. n (n = 0 to 15)

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DRQC
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 7.53 DTFRRQCn Register Contents

Bit Position	Bit Name	Function
31 to 1	—	Reserved These bits are always read as 0. The write value should be 0.
0	DRQC	Hardware DMA transfer request clear If the hardware DMA transfer request is an edge detection type*1, a user can clear the DTFRRQn.DRQ bit by writing 1 to this bit. If the hardware DMA transfer request is a level input type*1, the DTFRRQn.DRQ bit cannot be cleared by writing to this bit. 0 is always read from this bit.

Note 1. Whether the hardware DMA transfer request is an edge detection type or level input type depends on the hardware DMA transfer source selected by DTFRn.REQSEL. The hardware DMA transfer request is only an edge detection type.

7.11 DTS Channel Register

7.11.1 Transfer information of the DTS (TI)

7.11.1.1 Structure of the TI

The transfer information of the DTS is called TI. One set of TI consists of 32 bits. 8 sets of TI are assigned for each channel. The 8 sets of TI is called TI-A, TI-B, TI-C, TI-D, TI- E, TI-F, TI-G, and TI-H.

Figure 7.12 shows the structure of the TI.

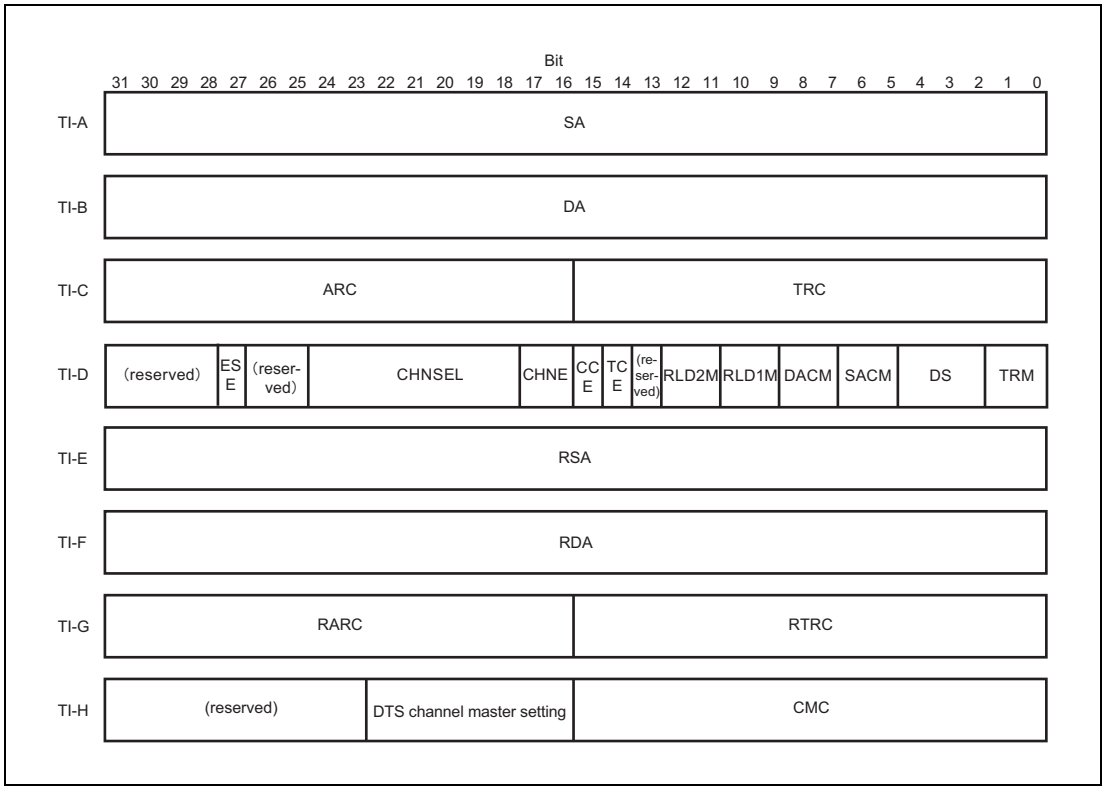


Figure 7.12 Structure of the TI

7.11.1.2 Organization of the TI in the DTSRAM

A user indirectly accesses the DTSRAM by way of the DTS channel registers for each channel and the DTS channel master setting registers.

Therefore, usually, a user does not have to think about the address organization of the TI in the DTSRAM. As an exception, when ECC error occurs while the DTSRAM is read, the address of the ECC error in the DTSRAM is stored to the DTSRAM error register 2 (DTSER2), one of the global registers. In order to know which channel and TI have generated the error, you need to understand the address organization of the TI in the DTSRAM.

Figure 7.13 shows the address organization of the TI in the DTSRAM.

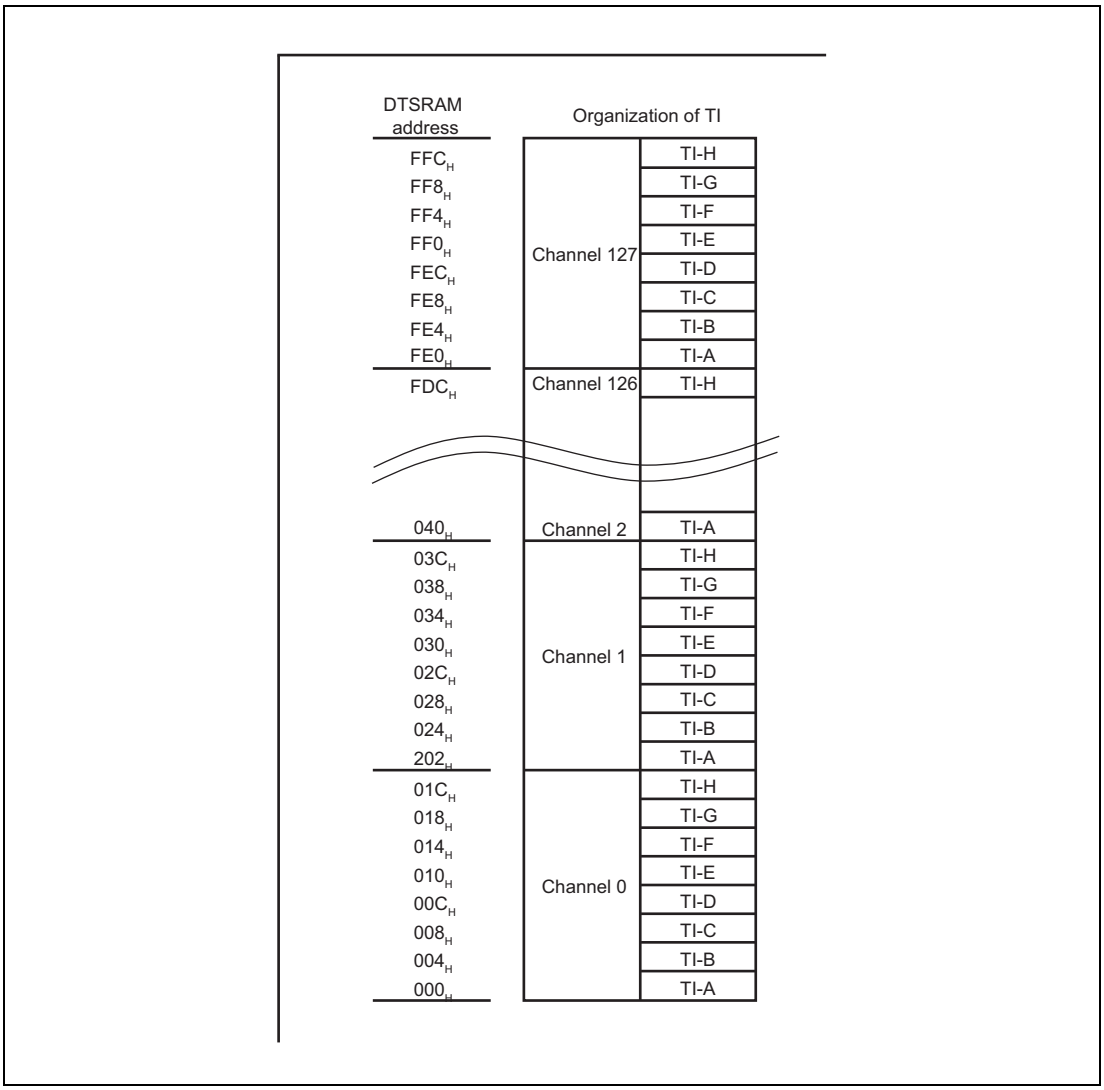


Figure 7.13 Organization of the TI in the DTSRAM

7.11.1.3 Accessing the TI

TI-A can be accessed by way of the DTS source address register (DTSAnnn) for each channel.

TI-B can be accessed by way of the DTS destination address register (DTDAnnn) for each channel.

TI-C can be accessed by way of the DTS transfer count register (DTTCnnn) for each channel.

TI-D can be accessed by way of the DTS transfer control register (DTTCTnnn) for each channel.

TI-E can be accessed by way of the DTS reload source address register (DTRSAnnn) for each channel.

TI-F can be accessed by way of the DTS reload destination address register (DTRDAnnn) for each channel.

TI-G can be accessed by way of the DTS reload transfer count register (DTRTCnnn) for each channel.

TI-H can be accessed by way of the channel master setting register (DTSnnnCM), which is a global register, and the transfer count compare register (DTTCCnnn) for each channel.

7.11.1.4 Caution about accessing the TI

The data of the DTS channel master setting register and the data of the DTS transfer count compare register are stored to the same TI-H.

Access to the DTS channel master setting register (DTSnnnCM) is actually 32-bit access to the whole TI-H. Therefore, when you write to the DTS channel master setting register, the lower 16-bit data, which is the data for the DTS transfer count compare (CMC), is updated at the same time. When you read from the DTS channel master setting register, the value of the DTS transfer count compare (CMC) is read as the lower 16-bit data.

When you read from the DTS transfer count compare register (DTTCCnnn), 32-bit data is read from the TI-H, but only the lower 16-bit data is actually seen in the result of the register read. When you write to the DTS transfer count compare register (DTTCCnnn), lower 16-bit read/modify/write access to the 32-bit TI-H is used. Data in the TI immediately after the reset is undefined. It should be noted that, if you try to write to the DTS transfer count compare register (DTTCCnnn) before setting up the DTS channel master setting register, ECC error may be detected during the read of the read/modify/write access.

Bits 31 to 23 of the TI-H are not used, but you can read and write those bits by accessing the DTS channel master setting register. Our recommendation is as follows. When you write to those bits, write 0. When you read from those bits, discard the read value by software.

After the reset, the values in the DTSRAM, which stores the TI, are undefined. After the reset, if you read TI before you write to the TI, ECC error is generated. Therefore, the first access to the following registers after the reset must be write access. The first access after the reset should never be read access.

- DTS source address register (DTSAnnn)
- DTS destination address register (DTDAnnn)
- DTS transfer count register (DTTCnnn)
- DTS transfer control register (DTTCTnnn)
- DTS reload source address register (DTRSAnnn)
- DTS reload destination address register (DTRDAnnn)
- DTS reload transfer count register (DTRTCnnn)
- Channel master setting registers (DTSnnnCM)

In addition, after the reset, the first access to the DTS transfer count compare register (DTTCCnnn) must be always after write access to the channel master setting register (DTSnnnCM).

You can access the TI from a CPU while the DTS is executing DMA transfer. But if you do so, the following should be noted.

- While a channel is executing DMA transfer, the TI of the channel should not be updated by TI access from a CPU. If this situation happens, the result of the DMA transfer and the contents of the TI may mismatch.
- If TI access is requested from a CPU while a TI fetch or TI write back is executed, the TI access is executed after the completion of the TI fetch or TI write back. If a TI fetch or TI write back is requested while a TI access request from a CPU is processed, the TI fetch or TI write back is executed after the processing of the TI access is complete.

7.11.2 DTS Channel Register Address

Address = Base address "FFFF 9000_H" + Offset address

Offset Address	Register Symbol	Meaning	Accessed Permission	
			Special Master	General Master
$0000_H + 40_H \times$ [channel number]	DTSAnnn	DTS source address	√	√
$0004_H + 40_H \times$ [channel number]	DTDAnnn	DTS destination address	√	√
$0008_H + 40_H \times$ [channel number]	DTTCnnn	DTS transfer count	√	√
$000C_H + 40_H \times$ [channel number]	DTTCTnnn	DTS transfer control	√	√
$0010_H + 40_H \times$ [channel number]	DTRSAnnn	DTS reload source address	√	√
$0014_H + 40_H \times$ [channel number]	DTRDAnnn	DTS reload destination address	√	√
$0018_H + 40_H \times$ [channel number]	DTRTCnnn	DTS reload transfer count	√	√
$001C_H + 40_H \times$ [channel number]	DTTCCnnn	DTS transfer count compare	√	√
$0020_H + 40_H \times$ [channel number]	DTFSLnnn	DTSFSL operation setting	√	√
$0024_H + 40_H \times$ [channel number]	DTFSTnnn	DTSFSL transfer request status	√	√
$0028_H + 40_H \times$ [channel number]	DTFSSnnn	DTSFSL transfer request set	√	√
$002C_H + 40_H \times$ [channel number]	DTFSCnnn	DTSFSL transfer request clear	√	√

Note: The [channel number] in the offset addresses is a number in the range from 0 to 127.
The "nnn" in the register symbols is a 3-digit number in the range from 000 to 127.

7.11.3 Details of DTS Channel Registers

The “nnn” in the register symbols indicates the DTS channel number (nnn = 000 to 127).

7.11.3.1 DTSAnnn — DTS Source Address Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 9000_H + 40_H × Ch. No. n (n = 0 to 127)

Value after reset: Undefined

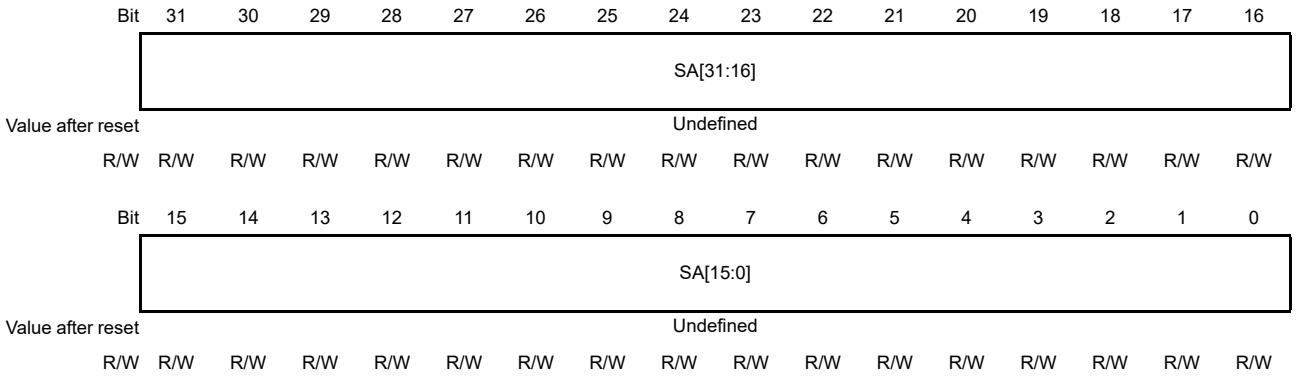


Table 7.54 DTSAnnn Register Contents

Bit Position	Bit Name	Function
31 to 0	SA[31:0]	Source address Specifies the DMA transfer source address. SA[31:0] are updated at the timing of the TI write back and retains the DMA transfer source address for the next DMA transfer.

CAUTION

DMA transfer for misaligned data is not supported. The possible lowest four bits of the address for each transfer data size is as follows. (× denotes an arbitrary one bit.) The correct operation is not guaranteed if you select other setting.

Data Size	SA3	SA2	SA1	SA0
8 bits	×	×	×	×
16 bits	×	×	×	0
32 bits	×	×	0	0
64 bits	×	0	0	0
128 bits	0	0	0	0

7.11.3.2 DTDAnnn — DTS Destination Address Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 9004_H + 40_H × Ch. No. n (n = 0 to 127)

Value after reset: Undefined

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DA[31:16]															
Value after reset	Undefined															
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DA[15:0]															
Value after reset	Undefined															
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.55 DTDAnnn Register Contents

Bit Position	Bit Name	Function
31 to 0	DA[31:0]	Destination address Specify the DMA transfer destination address. DA[31:0] are updated at the timing of the TI write back and retains the DMA transfer destination address for the next DMA transfer.

CAUTIONS

1. **If DMA transfer error is generated in the read cycle of DMA transfer, the write cycle is not executed, but the destination address is updated.**
2. **DMA transfer for misaligned data is not supported. The possible lowest four bits of the address for each transfer data size is as follows. (× denotes an arbitrary one bit.) The correct operation is not guaranteed if you select other setting.**

Data Size	DA3	DA2	DA1	DA0
8 bits	×	×	×	×
16 bits	×	×	×	0
32 bits	×	×	0	0
64 bits	×	0	0	0
128 bits	0	0	0	0

7.11.3.3 DTTCnnn — DTS Transfer Count Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 9008_H + 40_H × Ch. No. n (n = 0 to 127)

Value after reset: Undefined

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ARC[15:0]															
Value after reset	Undefined															
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TRC[15:0]															
Value after reset	Undefined															
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.56 DTTCnnn Register Contents

Bit Position	Bit Name	Function
31 to 16	ARC[15:0]	Address reload count Specify the number of transfers until the address reload when the reload function 2 is used, and also specifies the number of transfers when the block transfer 2 is used. When the reload function 2 or block transfer 2 is used, ARC[15:0] are decremented by one for every DMA cycle and updated at the timing of the TI write back. When the reload function 2 or block transfer 2 is not used, ARC[15:0] are not updated. If 0000_H is set, address reload is disabled. If the value at the start of a DMA cycle is 0000_H, subtraction from the address reload count does not proceed even after the DMA cycle.
15 to 0	TRC[15:0]	Transfer count Configure the number of transfers. TRC[15:0] are decremented by one whenever a DMA cycle is executed. It is updated at the timing of the TI write back. If the reload function is not used, after the last transfer is complete, the value at the completion (0000_H) is retained. If 0000_H is set, DMA transfer is not executed even when a DMA transfer request is accepted.

TRC[15:0]	Operation
0000_H	Transfer is disabled or complete.
0001_H	The number of transfers or remaining transfers is 1.
:	:
$FFFF_H$	The number of transfers or remaining transfers is 65535.

CAUTIONS

1. **If transfer error is generated in the read cycle of DMA transfer, the write cycle is not executed, but the transfer count and the address reload count are updated.**
2. **Unlike a DMAC, "0000_H" in the transfer count of the DTS does not mean "65536 transfers" but means that transfer is disabled or complete.**

7.11.3.4 DTTCTnnn — DTS Transfer Control Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 900C$_H$ + 40$_H$ × Ch. No. n (n = 0 to 127)

Value after reset: Undefined

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	ESE	—	—	CHNSEL[6:0]							CHNE[1:0]	
Value after reset	Undefined															
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CCE	TCE	—	RLD2M[1:0]		RLD1M[1:0]		DACM[1:0]		SACM[1:0]		DS[2:0]			TRM[1:0]	
Value after reset	Undefined															
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.57 DTTCTnnn Register Contents (1/3)

Bit Position	Bit Name	Function
31 to 28	—	Reserved When read, the value returned is undefined. The write value should be 0.
27	ESE	Transfer error case DMA transfer abort setting Specifies whether to abort DMA transfer when DMA transfer error is generated. If this bit is cleared to 0, DMA transfer continues when DMA transfer error is generated. If this bit is set to 1, DMA transfer is aborted when DMA transfer error is generated. 0: DMA transfer continues when DMA transfer error is generated. 1: DMA transfer is aborted when DMA transfer error is generated.
26, 25	—	Reserved When read, the value returned is undefined. The write value should be 0.
24 to 18	CHNSEL[6:0]	Next channel in the chain Specifies the next channel in the chain. The next channel must be another channel in the DTS. You cannot specify a channel in a DMAC. You cannot specify the same channel for the next channel. (If you do, the correct operation is not guaranteed.)
17, 16	CHNE[1:0]	Chain enable Selects the chain function. 00: Disabled 01: Chain at the last transfer A chain request is generated at the completion of the DMA cycle in which the remaining transfer count is one. 10: (Setting prohibited. Operation is not guaranteed if the setting is made.) 11: Always chain A chain request is generated at the completion of every DMA cycle.
15	CCE	Transfer count match interrupt enable If this bit is set, a transfer count match interrupt is generated at the completion of the DMA cycle in which the transfer count compare register and the remaining transfer count have the same value.
14	TCE	Transfer completion interrupt enable If this bit is set, a transfer completion interrupt is generated at the completion of the last transfer.
13	—	Reserved When read, the value returned is undefined. The write value should be 0.

Table 7.57 DTTCTnnn Register Contents (2/3)

Bit Position	Bit Name	Function
12, 11	RLD2M[1:0]	Reload function 2 setting Configures the reload function 2. 00: Reload function 2 is disabled. 01: Reload function 2 is enabled. The source address and address reload count are reloaded at the completion of the DMA cycle in which the address reload count is 1. 10: Reload function 2 is enabled. The destination address and address reload count are reloaded at the completion of the DMA cycle in which the address reload count is 1. 11: Reload function 2 is enabled. The source address, destination address, and address reload count are reloaded at the completion of the DMA cycle in which the address reload count is 1.
10, 9	RLD1M[1:0]	Reload function 1 setting Configures the reload function 1. 00: Reload function 1 is disabled. 01: Reload function 1 is enabled. The source address and transfer count are reloaded at the completion of the DMA cycle in which the remaining transfer count is 1. (If the reload function 2 is enabled, the address reload count is also reloaded.) 10: Reload function 1 is enabled. The destination address and transfer count are reloaded at the completion of the DMA cycle in which the remaining transfer count is 1. (If the reload function 2 is enabled, the address reload count is also reloaded.) 11: Reload function 1 is enabled. The source address, destination address, and transfer count are reloaded at the completion of the DMA cycle in which the remaining transfer count is 1. (If the reload function 2 is enabled, the address reload count is also reloaded.)
8, 7	DACM[1:0]	Destination address count direction Specifies the count direction of the destination address.
6, 5	SACM[1:0]	Source address count direction Specifies the count direction of the source address.
4 to 2	DS[2:0]	Transfer data size Specifies the transfer data size.

DACM1	DACM0	Direction of Counting
0	0	Increment
0	1	Decrement
1	0	Fixed
1	1	Setting prohibited (operation not guaranteed)

SACM1	SACM0	Direction of Counting
0	0	**Increment**
0	1	Decrement
1	0	Fixed
1	1	Setting prohibited (operation not guaranteed)

DS2	DS1	DS0	Transfer Data Size
0	0	0	8 bits
0	0	1	16 bits
0	1	0	32 bits
0	1	1	64 bits
1	0	0	128 bits
Other than the above			Setting prohibited (operation not guaranteed)

Table 7.57 DTTCTnnn Register Contents (3/3)

Bit Position	Bit Name	Function
1, 0	TRM[1:0]	Transfer mode Specifies the DMA transfer mode. 00: Single transfer 01: Block transfer 1 (The number of transfers is specified by the transfer count.) 10: Block transfer 2 (The number of transfers is specified by the address reload count.) 11: Setting prohibited (operation not guaranteed)

CAUTIONS

1. **If forbidden settings are used for some of the bits, the correct operation is not guaranteed.**
2. **Bits 31 to 28, 26, 25, and 13 are unused, but you can read and write those bits. Our recommendation is as follows. When you write to those bits, write 0. When you read from those bits, discard the read value.**

7.11.3.5 DTRSAnnn — DTS Reload Source Address Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 9010_H + 40_H × Ch. No. n (n = 0 to 127)

Value after reset: Undefined

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RSA[31:16]															
Value after reset	Undefined															
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RSA[15:0]															
Value after reset	Undefined															
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.58 DTRSAnnn Register Contents

Bit Position	Bit Name	Function
31 to 0	RSA[31:0]	Reload source address Specify the source address to be reloaded when the reload function 1 or reload function 2 is used.

CAUTION

DMA transfer for misaligned data is not supported. The possible lowest four bits of the address for each transfer data size is as follows. (× denotes an arbitrary one bit.) The correct operation is not guaranteed if you select other setting.

Data Size	RSA3	RSA2	RSA1	RSA0
8 bits	×	×	×	×
16 bits	×	×	×	0
32 bits	×	×	0	0
64 bits	×	0	0	0
128 bits	0	0	0	0

7.11.3.6 DTRDAnnn — DTS Reload Destination Address Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 9014_H + 40_H × Ch. No. n (n = 0 to 127)

Value after reset: Undefined

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RDA[31:16]															
Value after reset	Undefined															
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RDA[15:0]															
Value after reset	Undefined															
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.59 DTRDAnnn Register Contents

Bit Position	Bit Name	Function
31 to 0	RDA[31:0]	Reload destination address Specifies the destination address to be reloaded when the reload function 1 or reload function 2 is used.

CAUTION

DMA transfer for misaligned data is not supported. The possible lowest four bits of the address for each transfer data size is as follows. (× denotes an arbitrary one bit.) The correct operation is not guaranteed if you select other setting.

Data Size	RDA3	RDA2	RDA1	RDA0
8 bits	×	×	×	×
16 bits	×	×	×	0
32 bits	×	×	0	0
64 bits	×	0	0	0
128 bits	0	0	0	0

7.11.3.7 DTRTCnnn — DTS Reload Transfer Count Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 9018_H + 40_H × Ch. No. n (n = 0 to 127)

Value after reset: Undefined

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RARC[15:0]															
Value after reset	Undefined															
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RTRC[15:0]															
Value after reset	Undefined															
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.60 DTRTCnnn Register Contents

Bit Position	Bit Name	Function
31 to 16	RARC[15:0]	Reload address reload count Specifies the value to be reloaded to the address reload count when the reload function 2 is used.
15 to 0	RTRC[15:0]	Reload transfer count Specifies the value to be reloaded to the transfer count when the reload function 1 is used.

RTRC[15:0]	Operation
0000_H	No DMA transfer
0001_H	1 transfer
:	:
$FFFF_H$	65535 transfers

7.11.3.8 DTTCCnnn — DTS Transfer Count Compare Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 901C$_H$ + 40$_H$ × Ch. No. n (n = 0 to 127)

Value after reset: Undefined

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CMC[15:0]															
Value after reset	Undefined															
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 7.61 DTTCCnnn Register Contents

Bit Position	Bit Name	Function
31 to 16	—	Reserved These bits are always read as 0. The write value should be 0.
15 to 0	CMC[15:0]	Transfer count compare Configures the transfer count to be compared to the transfer count register. If the transfer count match interrupt enable (DTTCTnnn.CCE) bit in the DTS transfer control register is 1, a transfer count match interrupt is generated at the completion of the DMA cycle in which the remaining transfer count is the same as the value in this register. If 0000$_H$ is set, comparison with the transfer count is disabled. In this case, a transfer count match interrupt is not generated.

CAUTION

This register must be accessed after the DTS channel master setting register is set up.

If you access this register without setting up the DTS channel master setting register after the reset, ECC error may be generated during access to this register.

7.11.3.9 DTFSLnnn — DTSFSL Operation Setting Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 9020_H + 40_H × Ch. No. n (n = 0 to 127)

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	REQEN
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 7.62 DTFSLnnn Register Contents

Bit Position	Bit Name	Function
31 to 1	—	Reserved These bits are always read as 0. The write value should be 0.
0	REQEN	DMA transfer request enable This bit selects whether a DMA transfer request from this channel retained in the DTSFSL is used as a candidate in DTS channel arbitration. 0: A DMA transfer request from this channel is not used as a candidate in DTS channel arbitration. 1: A DMA transfer request from this channel is used as a candidate in DTS channel arbitration. If this bit is 0, even if the DTSFSL retains a DMA transfer request, this channel is not a candidate in DTS channel arbitration inside the DTSFSL, and consequently a DMA transfer request from this channel is not generated.

7.11.3.10 DTFSTnnn — DTSFSL Transfer Request Status Register

Access: This register can be read in 32-bit units.

Address: FFFF 9024_H + 40_H × Ch. No. n (n = 0 to 127)

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DRQ
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 7.63 DTFSTnnn Register Contents

Bit Position	Bit Name	Function
31 to 1	—	Reserved
0	DRQ	DMA transfer request pending This bit shows whether a DMA transfer request of this channel is pending. This bit is set when the hardware transfer source input is detected, or when software writes “1” to the DTFSSnnn.DRQ bit. This bit is automatically cleared when the DTS accepts the DMA transfer request while the DTSFSL is requesting DMA transfer by this channel. Alternatively, software can clear this bit by writing 1 to the DTFSCnnn.DRQC bit. 0: A DMA transfer request is not pending. 1: A DMA transfer request is pending.

7.11.3.11 DTFSSnnn — DTSFSL Transfer Request Set Register

Access: This register can be read/written in 32-bit units.

Address: FFFF 9028_H + 40_H × Ch. No. n (n = 0 to 127)

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DRQS
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 7.64 DTFSSnnn Register Contents

Bit Position	Bit Name	Function
31 to 1	—	Reserved These bits are always read as 0. The write value should be 0.
0	DRQS	DMA transfer request set A user can set the DTFSTnnn.DRQ bit by writing 1 to this bit. 0 is always read from this bit.

7.11.3.12 DTFSCnnn — DTSFSL Transfer Request Clear Register

Access: This register can be read/written in 32-bit units.

Address: FFFF $902C_H$ + 40_H × Ch. No. n (n = 0 to 127)

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DRQC
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 7.65 DTFSCnnn Register Contents

Bit Position	Bit Name	Function
31 to 1	—	Reserved These bits are always read as 0. The write value should be 0.
0	DRQC	DMA transfer request clear A user can clear the DTFSTnnn.DRQ bit by writing 1 to this bit. 0 is always read from this bit.

Section 8 Resets

8.1 Features

- The $\overline{\text{RES}}$ pin incorporates a noise canceller.
- The source of a reset can be determined by referring the reset source determination register.
- The CPU can assert reset by setting a register.

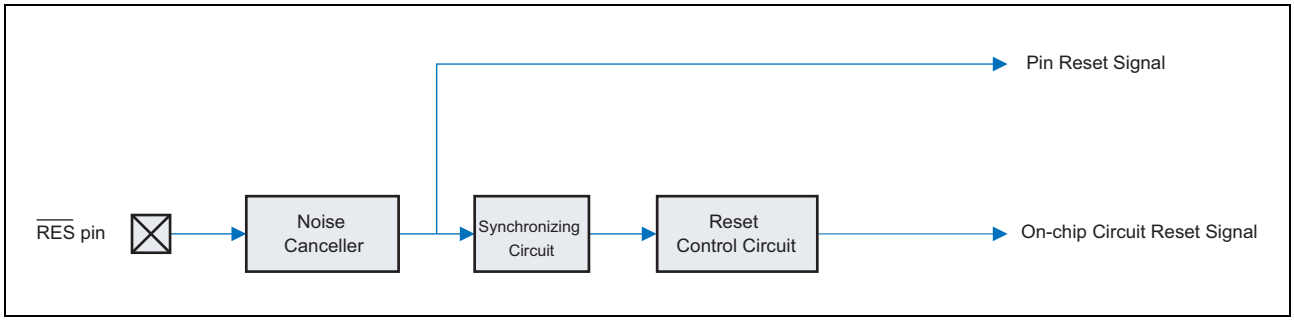


Figure 8.1 Reset Circuit

8.2 Reset State

8.2.1 External Reset State

When a low-level pulse longer than the noise cancel width (t_{RESNCW}) is input to the $\overline{RES}$ pin, external reset is received and this product transits to the external reset state. When external reset is received, individual pins transit to the external reset state. Refer to **Table 2.74, Pin State**, in **Section 2, Pins**, for the state of individual pins during external reset.

The low-level pulse width of the input signal must be longer than t_{RESW6} because the $\overline{RES}$ pin incorporates a noise cancelling circuit. This product transits to the internal reset state by driving the $\overline{RES}$ pin to the high-level after the required low-level period.

CAUTION

Refer to Section 37, Electrical Characteristics, for t_{RESNCW} and t_{RESW6}.

8.2.2 Internal Reset State

When a high-level signal longer than the noise cancel width (t_{RESNCW}) is input to the $\overline{RES}$ pin in the external reset state, this product transits to the internal reset state. Refer to **Table 2.74, Pin State**, in **Section 2, Pins**, for the state of individual pins.

The internal reset state is released in 200µs (max.) after external reset release, and the CPU starts the reset exception process.

CAUTION

Refer to Section 37, Electrical Characteristics for t_{RESNCW}.

8.3 Reset Sources

The following are the reset sources of this product.

Some registers are only initialized by the external reset state. That is, they are only initialized by input of the low level on the $\overline{\text{RES}}$ pin.

However, most registers are initialized by both external and internal reset states. This means that they are initialized by resets from any source.

Regarding sources for the initialization of registers, see the descriptions of the registers in the relevant sections. Where there is no explicit description of the sources for resetting and initializing a register, the register is initialized by both external and internal reset states. That is, such registers are initialized by resets from any source. However, the value after reset of some registers are undefined. Take care since the values of such registers following a reset are not fixed.

Source	Description
When the $\overline{\text{RES}}$ pin is driven to the low level	Transits to the external reset state.
When the reset request is asserted from ECM	Transits to the internal reset state.
When reset is asserted from the debugger (forced reset is issued)	Transits to the external reset state.
When the CPU sets the software reset request register	Transits to the internal reset state

8.4 Register Specifications

8.4.1 List of Registers

Table 8.1 List of Registers

Reset Source Determination

Address	Register Name	Description	Access Width	Value after Reset	Access Protection
FFF8 2800	RESF	Reset source determination register	32	0000 0000_H	
FFF8 2808	RESFC	Reset source clear register	32	0000 0000_H	

Power-On Detection

Address	Register Name	Description	Access Width	Value after Reset	Access Protection
FFF8 AC10	POF	Power-on clear flag register	32	0000 $000X_H$	
FFF8 AC14	POFC	Power-on clear flag clear register	32	0000 0000_H	

Software Reset Control

Address	Register Name	Description	Access Width	Value after Reset	Access Protection
FFF8 AC18	SWRESA	Software Reset Request Register	32	0000 0000_H	PROT1PHCMD

8.4.2 RESF — Reset Source Determination Register

This register determines the reset source.

The flags of this register can be cleared by the RESFC register and can be initialized by an external reset.

The register is initialized only by an external reset. It cannot be initialized by an internal reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RESF1	RESF0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 8.2 RESF Register Contents

Bit Position	Bit Name	Function
1	RESF1	ECM reset Indicates that a reset event occurred. 0: Event not occurred 1: Event occurred
0	RESF0	Software reset Indicates that a reset event occurred. 0: Event not occurred 1: Event occurred

8.4.3 RESFC — Reset Source Clear Register

This register clears the reset source indicated by the RESF register.

This register is always read as 00_H and can be initialized by an external reset.

The register is initialized only by an external reset. It cannot be initialized by an internal reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RESFC1	RESFC0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Table 8.3 RESFC Register Contents

Bit Position	Bit Name	Function
1	RESFC1	ECM reset Clears the status bit. 0: Do not clear 1: Clear
0	RESFC0	Software reset Clears the status bit. 0: Do not clear 1: Clear

8.4.4 POF — Power-On Clear Flag Register

This register determines whether SYSVCC detects the transition of the power from off to on. The POF flag becomes 1 (cold start) when SYSVCC detects the transition of the power from off to on.

Note that this register does not change even when “0” or “1” is written.

This register becomes 0 when 0 or 1 is written to the POFC bit of the POFC register.

The POF flag does not change even when other reset sources occur while the POF flag has been set to 0 (warm start can be determined).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	POF
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	x
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 8.4 POF Register Contents

Bit Position	Bit Name	Function
0	POF	Cold start event detection Indicates that a cold start event occurred. 0: Not occurred 1: Occurred

8.4.5 POFC — Power-On Clear Flag Clear Register

This register clears the POF register. This register can be reset by either an internal or external reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	POFC
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 8.5 POFC Register Contents

Bit Position	Bit Name	Function
0	POFC	The POF bit in the POF register is cleared to 0 when 0 or 1 is written to the POFC bit in the POFC register.

8.4.6 SWRESA — Software Reset Request Register

This register generates an internal reset when accessed. A software reset is issued when 1 is written to SWRESA. This register can be protected by the PROT1PHCMD register. This register can be reset by either an internal or external reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	SWRESA
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 8.6 SWRESA Register Contents

Bit Position	Bit Name	Function
0	SWRESA	0: — (default) 1: Internal reset enabled (internal reset triggered)

Example) Sequence of writing to register SWRESA

Writing to register SWRESA is only possible with the protection unlock sequence described in Section 11, Clock Controller.

Procedure

1. Write the fixed value (0000 $00A5_H$) to register PROT1PHCMD.
2. Write the setting value (0000 0001_H) to register SWRESA.
3. Write the inverse (FFFF $FFFE_H$) to register SWRESA.
4. Write the new setting value (0000 0001_H) to register SWRESA.

The setting value (0000 0001_H) can only be written to register SWRESA by following the procedure above.

Protection is not unlocked if the procedure is not followed correctly, the setting value (0000 0001_H) is not written to register SWRESA, and the PROT1PS.PROTERR bit is set to 1.

In case of failure to follow the sequence, repeat the procedure from the beginning.

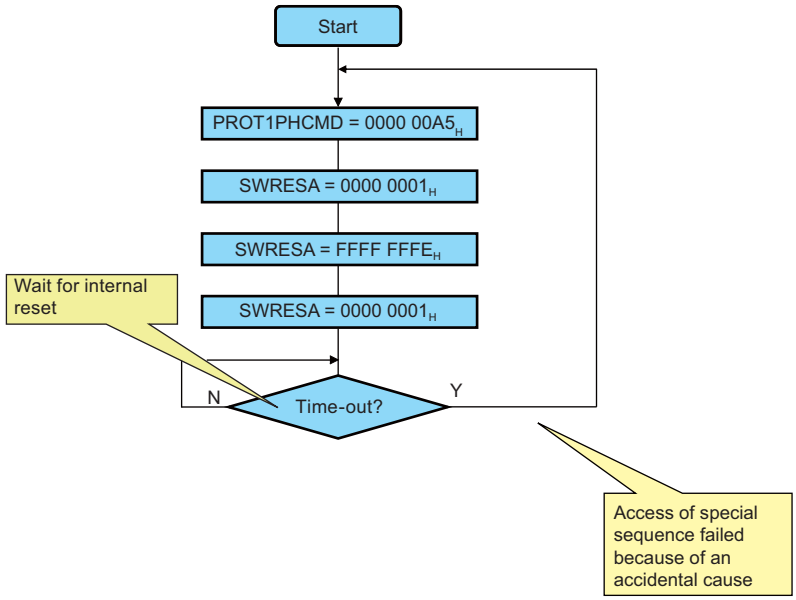
In case of writing to another register during steps 1 to 4 of the sequence, the protection function operates as follows. Operation is also as described when any interrupt is accepted during the sequence and the interrupt handling includes access to another register.

- When the writing is to another register in the same module*1, writing to the protected register fails and the PROT1PS.PROTERR bit is set to 1.
- When writing is to a register in another module, writing to the protected register will successfully proceed to completion, if the remainder of the procedure is followed correctly.

Reading of another register while the sequence is in progress does not lead to failure.

Note 1. For the applicable registers, see Section 11.4.10, PROT1PHCMD — Protect 1 Command Register.

8.5 Software Reset



Note: Timeout time should be 5 µs or longer.

Figure 8.2 Flow of Software Reset

8.6 RAM Retention

The values of RAM in retention area are not destroyed by reset.

8.7 Usage Notes

The debug function includes a reset masking function. To prevent unexpected operations of this function, fix the $\overline{\text{TRST}}$ pin to the low level when the debug function is not used.

Section 9 Power Supply Circuit

9.1 Features

- External Pass Transistor (EPT) method

With EPT	Without EPT
Operable with 2-output power management IC (5 V, 3.3 V)	Operable with 3-output power management IC (5 V, 3.3 V, 1.25 V)

- System power supply pins (SYSVCC)
  In the standby state, SYSVCC supplies power to the system logic and retention RAM.

Power Supply Name for Power Supply Pin	Pin Name	Power Supply Voltage in Operation	Uses of Power Supply
SYSVCC		3.0 V to 3.6 V	Power for system logic and retained RAM
VCC		3.0 V to 3.6 V	Power for oscillator, EPT, and flash programming
PLLVCC		3.0 V to 3.6 V	PLL
VDD		1.15 V to 1.35 V	Core power (direct power supply) and stabilizing capacity for core power
EVCC		4.5 V to 5.5 V	Port (5 V)
LVDVCC		3.0 V to 3.6 V	MSC (VccLVDS)
TTLVCC		3.0 V to 3.6 V/4.5 V to 5.5 V	MSC (VccTTL), Port (5 V, 3.3 V)
A0VCC/A1VCC		4.5 V to 5.5 V	Power for SAR AD
	A0VREFH/ A1VREFH	4.5 V to 5.5 V	Reference voltage for SAR AD
ADSVCC		4.5 V to 5.5 V	Power for ΔΣ AD
	ADSVREFH	4.5 V to 5.5 V	Reference voltage for power for ΔΣ AD
	EPTVOUT	—	EPT control for VDD **Note: The voltage is not externally applied.**
	RAMVCL	—	Stabilizing capacity for retained RAM
	ADSVCL	—	Stabilizing capacity for power for ΔΣ AD

For the relationship between the names of power supply pins and the pins, see **Section 37.2.1, Relationship between Power Name and Pin**.

9.2 Example of Connection of Power Management IC

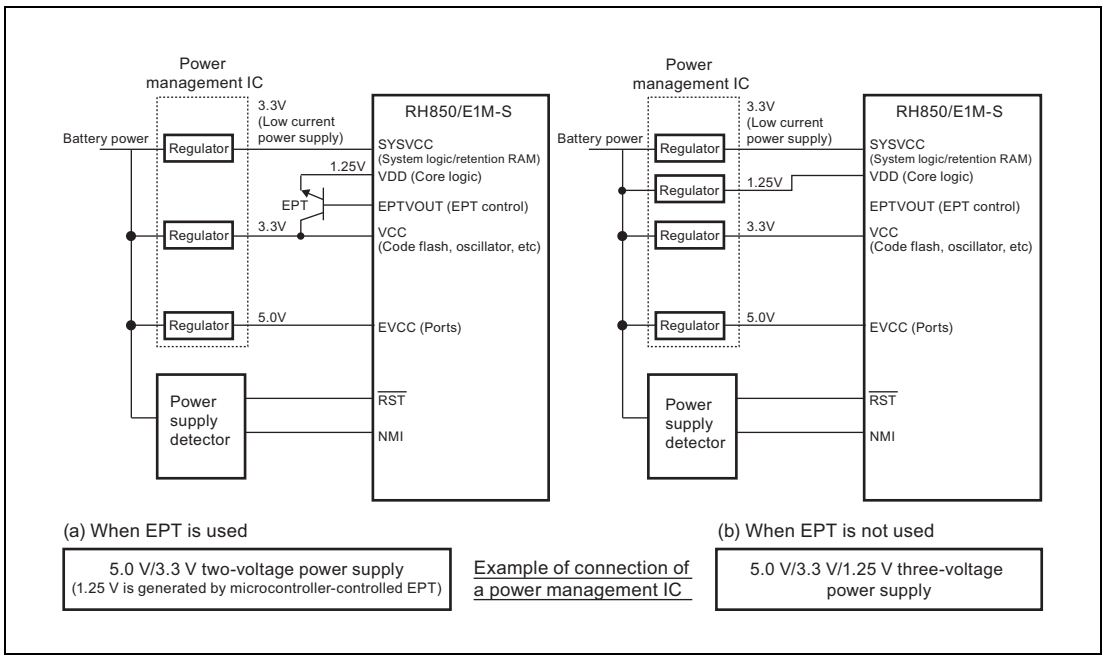


Figure 9.1 Power: Examples of Connection of a Power Management IC

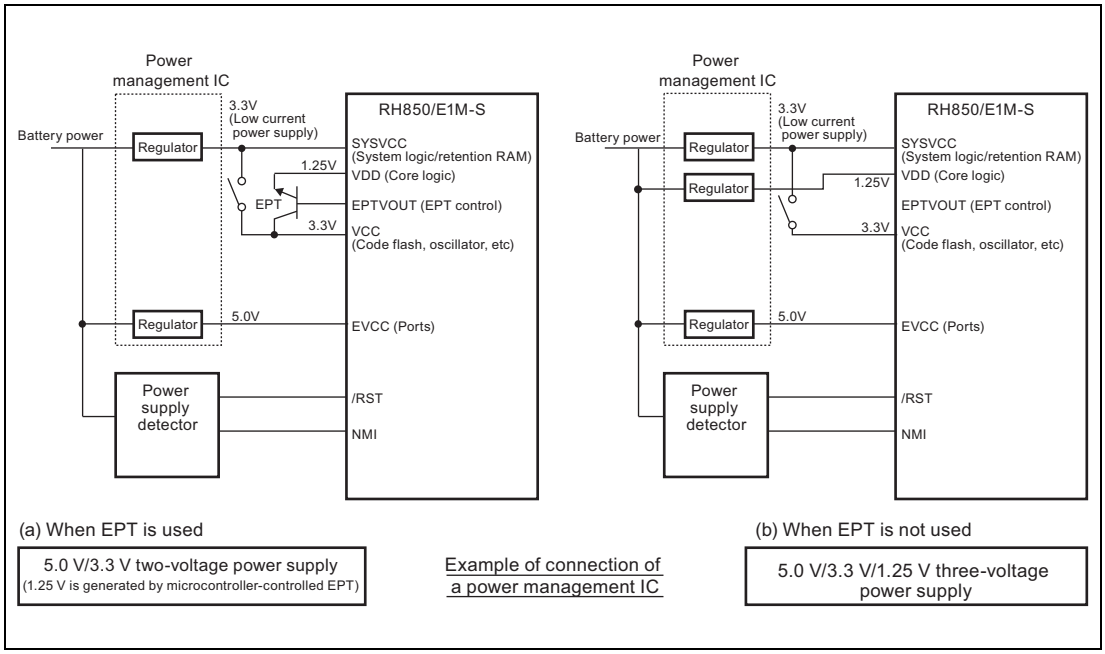


Figure 9.2 Power: Examples of Connection of a Power Management IC, Switched Off

9.3 Power-on Sequence

For details, see **Section 37, Electrical Characteristics**.

9.4 Usage Notes

Handling of power supply pins are described in this section. For handling of unused pins, see **Section 2.5.4, Handling of Unused Pins**.

Power Supply Name for Power Supply Pin	Pin Name	Required Setting	Recommended Handling
SYSVCC		Decoupling capacity: 0.1µF ±30% in total or higher	
VCC		Stabilizing capacity for power: 10µF ±30% or higher	Decoupling capacitor for each pin
PLLVCC			Shared with VCC Decoupling capacitor for each pin
VDD			Same as Guide to Mounting on Boards with an EPT in the following section
EVCC			Connect an external capacitor to suit the conditions of usage.
LVDVCC			Connect an external capacitor to suit the conditions of usage.
TTLVCC			Connect an external capacitor to suit the conditions of usage.
A0VCC/ A1VCC			See the application note separately issued.
	A0VREFH /A1VREFH		See the application note separately issued.
ADSVCC			See the application note separately issued.
	ADSVREFH		See the application note separately issued.
	EPTVOUT	See **Section 2.5.4, Handling of Unused Pins** when EPT is not used. See **Section 9.5, Guide to Mounting on Boards with an EPT** when EPT is used.	
	RAMVCL	Stabilizing capacity for power: 0.1µF ±30%	
	ADSVCL		See the application note separately issued.

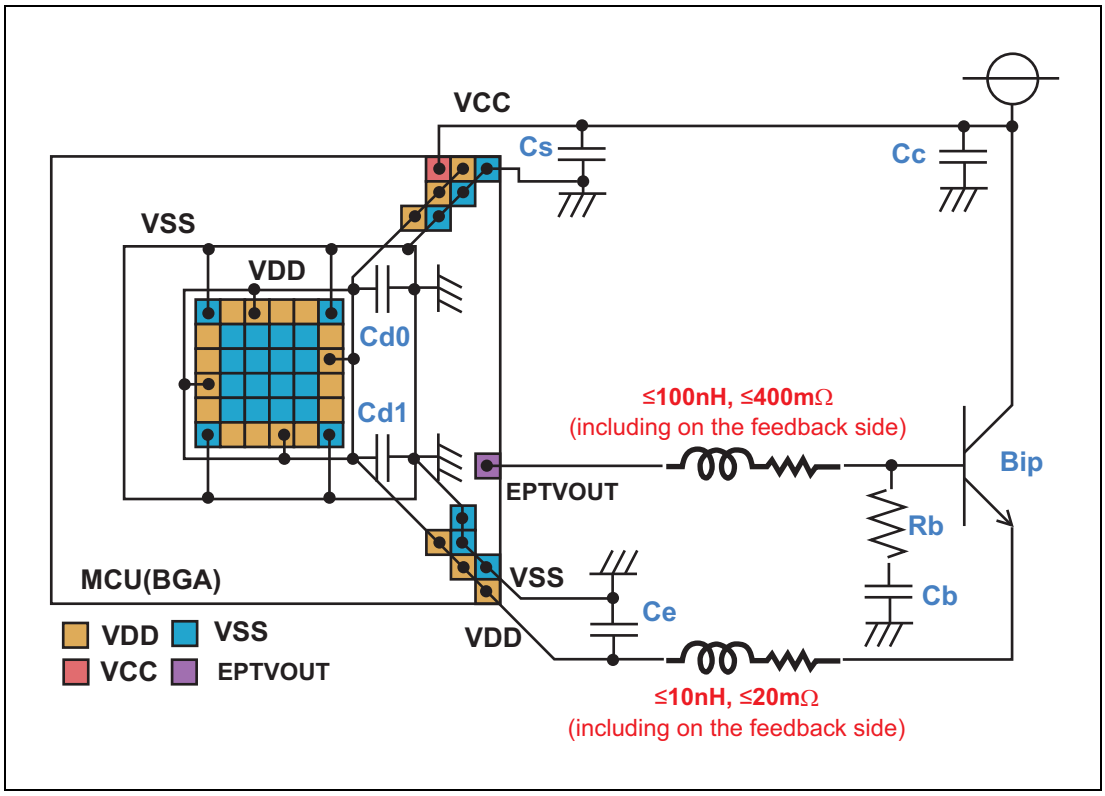
9.5 Guide to Mounting on Boards with an EPT

Comply with the guidelines below in mounting to obtain a VDD within the specified range if you are using a 2SD1899 npn bipolar transistor from Renesas as an EPT to handle the VDD power supply.

However, the guidelines specifically apply when the EPT is that stated above. Using a different transistor will require separate detailed consideration.

9.5.1 BGA

Placement of bypass capacitors: 2 × 2.2 µF if directly below the package and chip; 4.7 µF if drawn out externally to the package



Signal	Spec	Notes
Bip	—	NPN Bip. DC Gain 100 to 400 2SD1899 Renesas
Cc	≥ 5 µF	—
Rb	10Ω ±5%	Resistance tolerance: ±1%, Temperature characteristics: 200ppm/°C, Rated power: ≥ 0.05W
Cb	1 µF ±30%	Capacitance tolerance: ±10%, Temperature characteristics: X7R, X8R, ESR: < 50mΩ
Ce	4.7 µF ±30%	Capacitance tolerance: ±10%, Temperature characteristics: X7R, X8R, ESR: < 50mΩ Dimensions: 2012 *Arranged close to the package
Cdn (n: 0,1)	2.2 µF ±30% ×2	Capacitance tolerance: ±10%, Temperature characteristics: X7R, X8R, ESR: < 50mΩ Dimensions: 1608 *Two arranged directly below the package/chip
Cs	≥ 0.6 µF	ESR: < 50mΩ *Connected just proximal to the place between the VCC and VSS pins. If there is more than one VCC pin, the value in the left is the total capacity of these pins.

CAUTIONS

1. **Values for the parasitic L and R components produced by connection of VSS are based on the assumption that VSS is supplied through a plane of the board (with copper thickness of 35 µm and pitch of at least 1 mm for both VSS and VDD).**
2. **We recommend also forming the pattern for VDD as a plane. If this is impossible, make the wiring patterns as broad as is possible.**

9.5.2 Reference Information

Rough values of the parasitic L and R components for printed circuit boards are listed below.

Width of Wiring Runs (mm)	Inductance (nH/m)	Resistance (mΩ/m)
0.7	621	1210
1	555	913
2	432	571
4	263	257
6	200	183

CAUTION

The values are based on the assumption that VSS is supplied through a plane of the board (with copper thickness of 35 µm and pitch of at least 1 mm for both VSS and VDD).

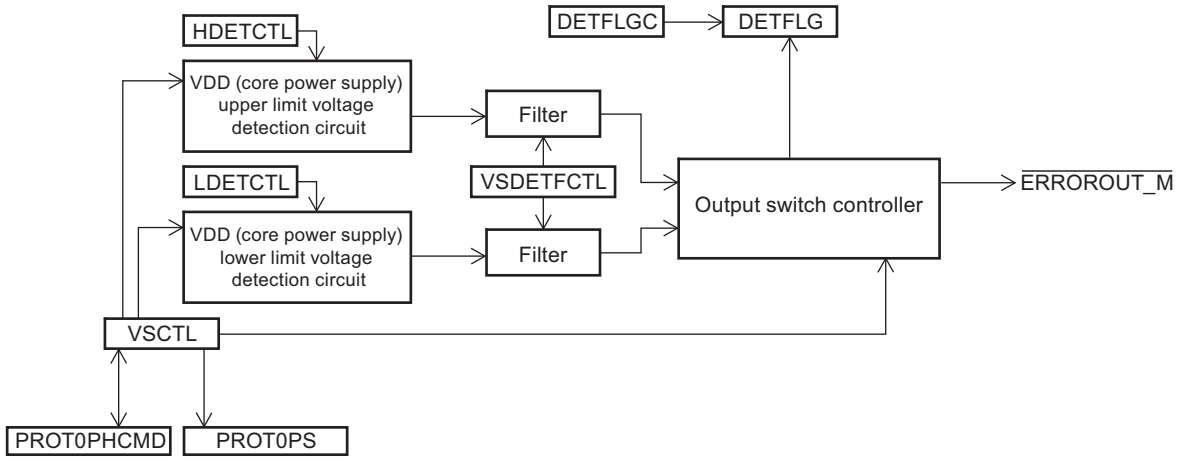
Section 10 Power Supply Voltage Monitor

10.1 Features

- The power supply voltage monitor is used for monitoring the VDD electric potential supplied to the core logic.
- The power supply voltage monitor has H- and L-side voltage detectors, which detect if the VDD is more or less than the specified voltage.
- Writing to the registers which select run or stop of the power supply voltage monitor function should be made in a specific sequence so that they cannot be incorrectly rewritten because of program malfunction and the like.
- If the power supply voltage monitor detects an error, it gives the error notice to external devices outside of LSI.

10.2 Configuration

Figure 10.1 shows the block diagram of the power supply voltage monitor.



Note 1. See **Section 10.3, Register Specifications** for the registers shown in the diagram above.

Figure 10.1 Block Diagram of the Power Supply Voltage Monitor

10.3 Register Specifications

Table 10.1 shows the specifications of registers.

Table 10.1 Register Specifications

Register Name	Abbreviation	R/W	Value After Reset	Address	Access Size
CVM detection flag register	DETFLG	R	0000 0000_H	FFF8 2820_H	32
CVM detection flag clear register	DETFLGC	R/W	0000 0000_H	FFF8 2828_H	32
CVM control register	VSCTL	R/W	0000 0000_H	FFF8 $2C10_H$	32
Upper limit voltage setting register	HDETCTL	R/W	0000 0003_H	FFF8 $2C14_H$	32
Lower limit voltage setting register	LDETCTL	R/W	0000 0003_H	FFF8 $2C18_H$	32
Detection signal filter control register	VSDETFCTL	R/W	0000 0000_H	FFF8 $2C1C_H$	32

10.3.1 DETFLG — CVM Detection Flag Register

This register indicates the detection state of upper/lower limit voltage. This register can only be reset by an external reset signal. This register also can be reset by the DETFLGC register.

Access: This register can be read in 32-bit units.

Address: FFF8 2820$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	LDETFLG	HDETFLG
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 10.2 DETFLG Register Contents

Bit Position	Bit Name	Function
1	LDETFLG	Lower Limit Voltage Detection Flag 0: Lower limit voltage has not been detected. 1: Lower limit voltage has been detected.
0	HDETFLG	Upper Limit Voltage Detection Flag 0: Upper limit voltage has not been detected. 1: Upper limit voltage has been detected.

10.3.2 DETFLGC — CVM Detection Flag Clear Register

This register clears the DETFLG register.

Access: This register can be read/written in 32-bit units.
Reading this register always returns 0000 0000_H.

Address: FFF8 2828_H

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	LDETFLGC	HDETFLGC
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Table 10.3 DETFLGC Register Contents

Bit Position	Bit Name	Function
1	LDETFLGC	Lower Limit Voltage Detection Flag Clear 0: Lower limit voltage detection flag is not cleared. 1: Lower limit voltage detection flag is cleared.
0	HDETFLGC	Upper Limit Voltage Detection Flag Clear 0: Upper limit voltage detection flag is not cleared. 1: Upper limit voltage detection flag is cleared.

10.3.3 VSCTL — CVM Control Register

This register enables or disables the CVM. This register is protected by the PROT0PHCMD register.

See the description of the PROT0PHCMD register for the details.

This register can only be reset by an external reset signal.

Access: This register can be read/written in 32-bit units.

Address: FFF8 2C10$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	EOUTEN	VSHEN	VSLEN
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Table 10.4 VSCTL Register Contents

Bit Position	Bit Name	Function
2	EOUTEN	0: Error output is masked. 1: Error output is not masked.
1	VSHEN	0: Upper limit voltage detection is disabled. 1: Upper limit voltage detection is enabled.
0	VSLEN	0: Lower limit voltage detection is disabled. 1: Lower limit voltage detection is enabled.

It takes 200 µs to start CVM operation after the VSHEN or VSLEN bit is enabled.

10.3.4 HDETCTL — Upper Limit Voltage Setting Register

This register controls the detection level of upper limit voltage. This register can only be reset by an external reset signal.

Access: This register can be read/written in 32-bit units.

Address: FFF8 2C14$_H$

Value after reset: 0000 0003$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VDDREFH1	VDDREFH0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Table 10.5 HDETCTL Register Contents

Bit Position	Bit Name	Function
1	VDDREFH1	Upper Limit Voltage Detection Level Setting These bits should always be set to 0 when used.
0	VDDREFH0	

VDDREFH1	VDDREFH0	Upper limit voltage detection level
0	0	1.50 V

Setting other than the above is prohibited.

10.3.5 LDETCTL — Lower Limit Voltage Register

This register controls the detection level of lower limit voltage. This register can only be reset by an external reset signal.

Access: This register can be read/written in 32-bit units.

Address: FFF8 2C18$_H$

Value after reset: 0000 0003$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	VDDREFL1	VDDREFL0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Table 10.6 LDETCTL Register Contents

Bit Position	Bit Name	Function
1	VDDREFL1	Lower Limit Voltage Detection Level Setting These bits should always be set to 0 when used.
0	VDDREFL0	

VDDREFL1	VDDREFL0	Lower limit voltage detection level
0	0	0.95 V

Setting other than the above is prohibited.

10.3.6 VSDETFCTL — Detection Signal Filter Control Register

This register controls the width of digital noise filter. This register can only be reset by an external reset signal.

Access: This register can be read/written in 32-bit units.

Address: FFF8 2C1C$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DETF1	DETF0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Table 10.7 VSDETFCTL Register Contents

Bit Position	Bit Name	Function
1	DETF1	Digital Noise Filter Width Setting
0	DETF0	

DETF1	DETF0	Digital noise filter width
0	**0**	Bypass the filter.
0	1	Filtering cycle: 16 clk
1	0	Filtering cycle: 32 clk
1	1	Filtering cycle: 64 clk

The sampling clock of the digital noise filter is ROSC (240 KHz).

10.4 Usage

The operation flow from CVM detection condition setting to actual detection is as follows:

1. Set the voltage conditions for H- and L-side voltage detection in the upper and lower voltage limit setting registers (HDETCTL and LDETCTL), respectively. Also set the voltage detection noise filter width in the detection signal filter control register, and complete to set all CVM-related registers. The CVM's H- and L-side enable bits (VSHEN and VSLEN) of the CVM control register (VSCTL) and the error notice to external devices outside of LSI (EOUTEN) are then specified to start operation.
2. When VDD exceeds the specified upper limit voltage, an error signal is output to external devices outside of LSI according to the settings of the EOUTEN bit in the VSCTL register and the upper limit voltage detection flag (the HDETFLG bit in the DETFLG register) is set. When VDD falls below the specified lower limit voltage, an error signal is output to external devices outside of LSI according to the settings of the EOUTEN bit in the VSCTL register and the lower limit voltage detection flag (the LDETFLG bit in the DETFLG register) is set.
3. Check the upper and lower limit voltage detection flags after H- and L-side voltage detection to verify the details of the corresponding detection.

The $\overline{\text{ERROROUT\_M}}$ pin (master) of the error control module (ECM) is also used in common for the CVM error output pin.

When $\overline{\text{ERROROUT\_M}}$ pin output is used in CVM, make related settings under the following procedures.

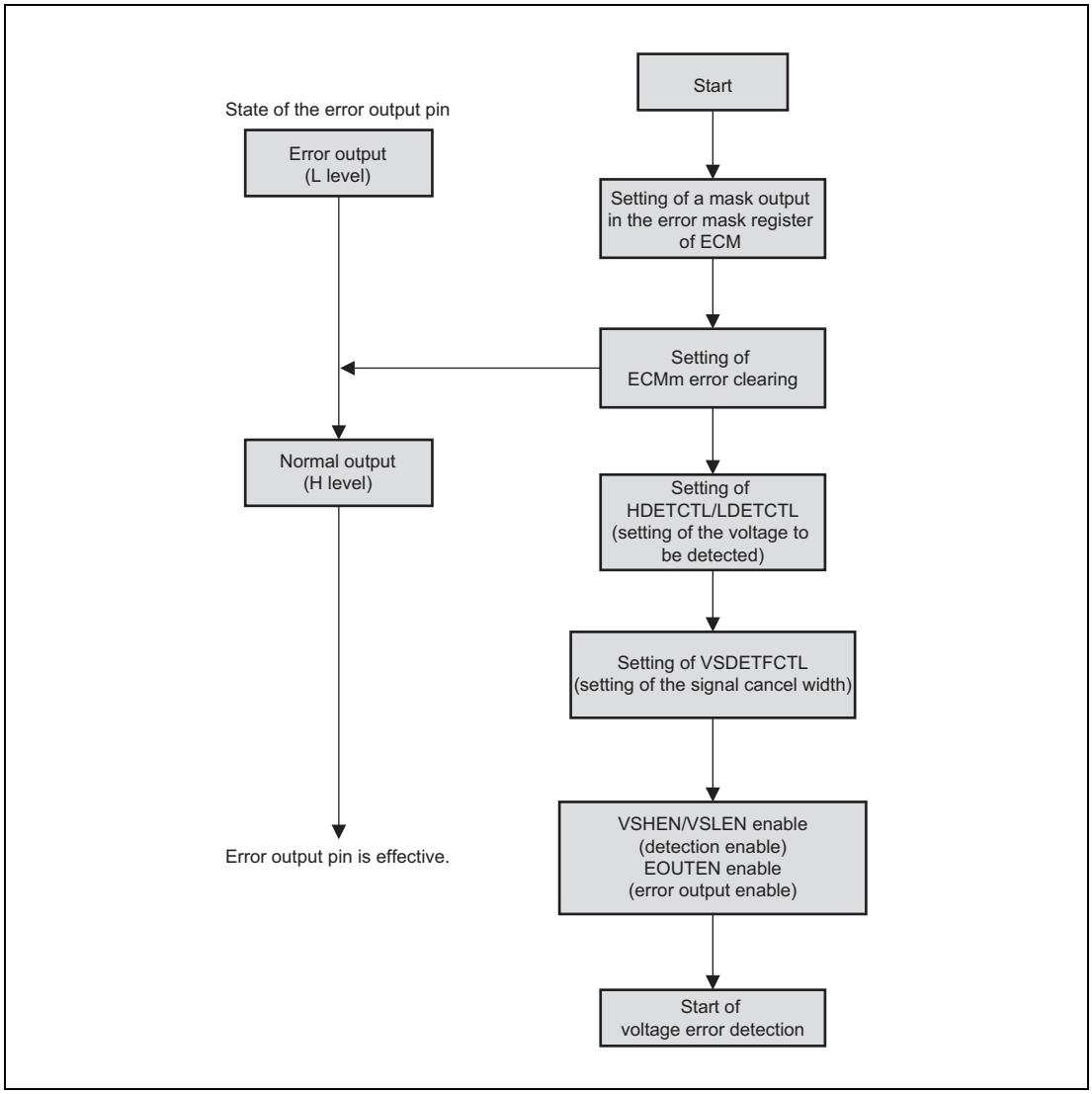


Figure 10.2 Flowchart

10.5 Notes on Usage

- The power supply voltage monitor operates only after release from an external reset. The power supply voltage monitor operation is stopped during an external reset, and various registers are initialized.
- The setting registers and flag registers are only initialized by an external reset. They cannot be initialized by an internal reset requested by the CPU or ECM.

Section 11 Clock Controller

11.1 Features

- An oscillation circuit is included (Main OSC).
- Oscillation at 20 MHz is possible without using external capacitors (only with limited resonators).
- A 20-MHz clock can be input from the EXTAL pin.
  A 20-MHz external clock can be directly input from the EXTAL pin, separately from the main OSC.

CAUTION

XTAL should be an open circuit when an external clock signal is being directly input.

- On-chip PLLs are included and can be used to divide clock frequencies.
- PLL0 is capable of operating as a spread spectrum clock generator (SSCG) for reducing radiated noise.
  The clock signal produced by PLL1 is not frequency modulated (i.e. is a clean clock signal) and so is suitable for use with timers and communications modules.
- The ring oscillator circuits (ROSC) is included.
- Interrupts can be generated when the Main OSC goes outside the frequency range specified by the clock monitor.
- The clock frequencies can be increased stepwise by software for suppressing inrush currents after release from the reset state.
  The division ratios for the CPU clock and the peripheral clocks can be selected with register settings (1/4, 1/2, and 1/1).
- A frequency divided clock set by a 1-MHz clock and a baud rate generator (BRG) can be output from the CK pin.

11.2 Type of Clocks

Table 11.1 shows the list of clocks, **Table 11.2** shows the operation clocks of each functional module, and **Figure 11.1** shows the block diagram of clocks.

Table 11.1 List of Clocks

When operating at 20 MHz input frequency (Main OSC)

Clock Name	Symbol	Clock Frequency: Division Ratio of Divider 0A/ Divider 1A 1/1	1/2	1/4	Remarks
CPU clock	CLK_CPU	320 MHz*1	160 MHz*1	80 MHz*1	PLL0 (SSCG can be selected)
		240 MHz*2	120 MHz*2	60 MHz*2	
High speed peripheral clock	CLK_HSB	80 MHz	40 MHz	20 MHz	
Low speed peripheral clock (peripheral clock)	CLK_LSB	40 MHz	20 MHz	10 MHz	
Unmodulated high speed peripheral clock	CLKC_HSB	80 MHz	40 MHz	20 MHz	PLL1 (SSCG cannot be selected)
Unmodulated low speed peripheral clock	CLKC_LSB	40 MHz	20 MHz	10 MHz	
Low speed ROSC clock	CLK_ROSCL		240 KHz		
ASIC clock (1 MHz clock)	CLKA_ASIC		1 MHz		1/20 X Main OSC
WDTA counter clock	WDTACLKI		250 KHz		1/80 X Main OSC

Note 1. For the products with 320 MHz (max.) CPUs
Note 2. For the products with 240 MHz (max.) CPUs

Table 11.2 Clocks and Functional Modules

Clock Name	Functional Module Name
CPU clock	CPU
High speed peripheral clock	PCU, INTC, DMAC, DTS, CSIH, DFE, FlexRay
Low speed peripheral clock (peripheral clock)	ECM, CRC
Unmodulated high speed peripheral clock	RHSB, ΔΣAD, TSG2, TAPA, PIC
Unmodulated low speed peripheral clock	CAN, LIN, ATU-IV, APA, WDTA*1, OSTM, SCI, AD
Low speed ROSC clock	CLM
WDTA counter clock	WDTA*1

Note 1. The PCLK of the WDTA (the clock on the P-Bus) is the unmodulated low-speed peripheral clock and the operating clock of the WDTA is the WDTA counter clock.

The PCLK mentioned in the section of each function module means the clock frequency listed in the 1/1 division ratio in **Table 11.1**, which corresponds to the clock name listed in **Table 11.2**.

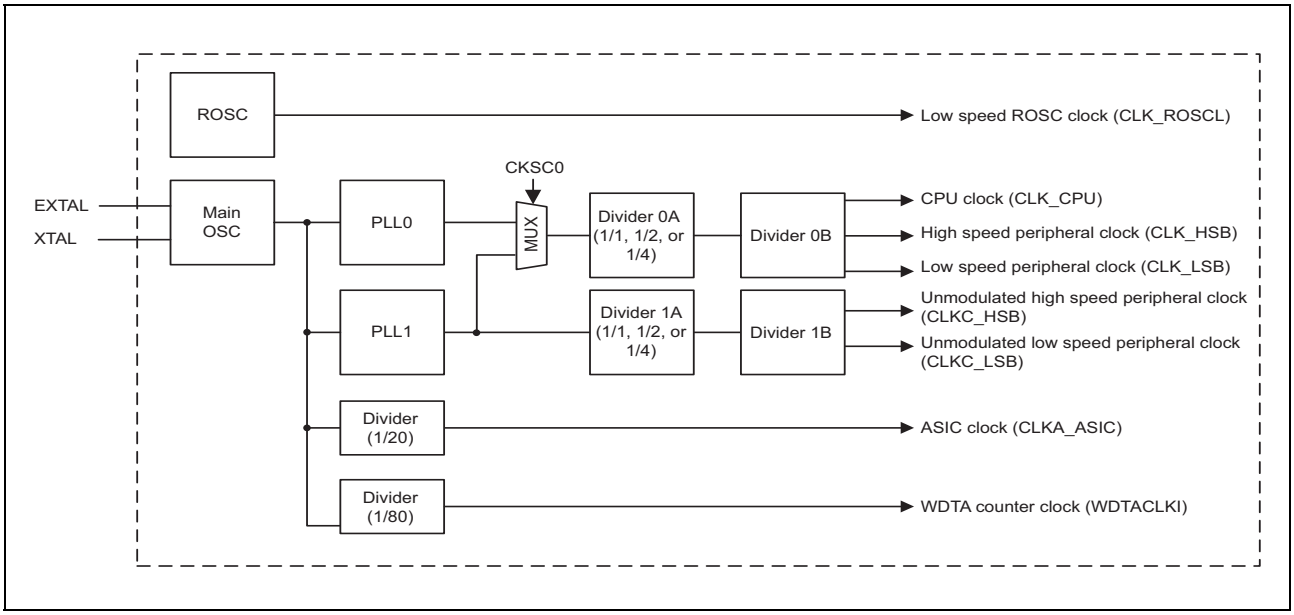


Figure 11.1 Block Diagram of Clock Controller

11.3 Input/Output Pins

Table 11.3 shows the pins related to the clock controller.

Table 11.3 Pins Related to the Clock Controller

Name	Pin Name	Input/Output	Function
External clock	EXTAL	Input	Input of a crystal resonator or external clock
Crystal	XTAL	Output	Connected to a crystal resonator
Clock output	CK	Output	1-MHz clock/BRG output
PLL power	PLLVCC	Input	Power supply for the PLL multiply circuit
PLL ground	PLLVSS	Input	Power supply for the PLL multiply circuit

11.3.1 How to Connect a Crystal Resonator

Figure 11.2 shows how to connect a crystal resonator. When the resonator recommended by the Company is used (the detail information is separately provided), any external component such as a load capacitor and a dumping resistor is not necessarily required for oscillation in general. However, proper operation should be verified under actual conditions prior to use.

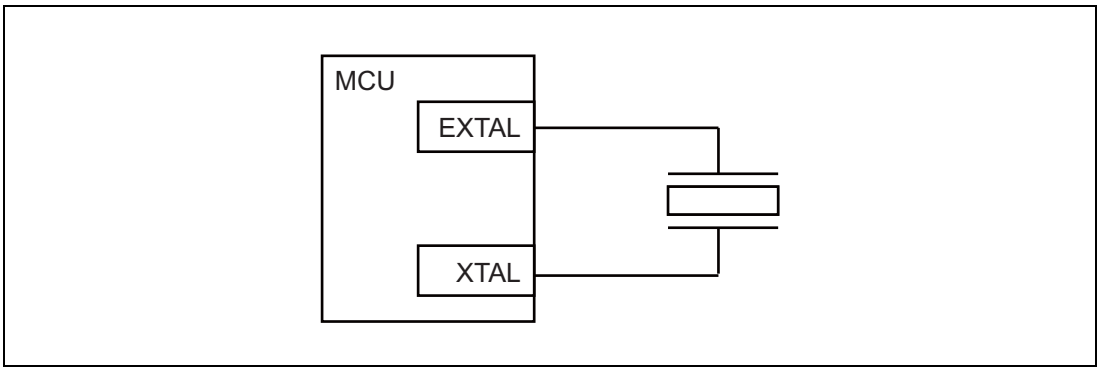


Figure 11.2 Connection Example of Crystal Resonator 1

11.4 Register Specification

11.4.1 List of Registers

Table 11.4 shows the list of registers.

Table 11.4 Registers Related to the Clock Controller

Register Name (abbreviation)	R/W	Value after Reset	Address	Access Size	Access Protection
PLL0 status register (PLL0CLKS)	R	0000 0001_H	FFF8 8004_H	32	
PLL0 control register 1 (PLL0CLKC1)	R/W	0000 0000_H	FFF8 8200_H	32	PROT1PHCMD
Clock 0 selection control register (CKSC0CTL)	R/W	0000 0020_H	FFF8 9000_H	32	PROT1PHCMD
Clock 0 selection active register (CKSC0ACT)	R	0000 0020_H	FFF8 9008_H	32	
Clock 0 division register (CLKD0DIV)	R/W	0000 0004_H	FFF8 8800_H	32	PROT1PHCMD
Clock 0 division status register (CLKD0STAT)	R	0000 0001_H	FFF8 8804_H	32	
Clock 1 selection control register (CKSC1CTL)	R/W	0000 0021_H	FFF8 9040_H	32	PROT1PHCMD
Clock 1 selection active register (CKSC1ACT)	R	0000 0021_H	FFF8 9048_H	32	
Protect 1 command register (PROT1PHCMD)	R/W	0000 0000_H	FFF8 $B000_H$	32	
Protect 1 status register (PROT1PS)	R	0000 0000_H	FFF8 $B004_H$	32	

11.4.2 PLL0CLKS — PLL0 Status Register

This register indicates the state of the PLL0 clock effective, active, or stable.

This register is initialized by either an internal or external reset.

Access: This register can be read in 32-bit units.

Address: FFF8 8004$_H$

Value after reset: 0000 0001$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	CLK ACT	CLK STAB	CLKEN
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0*1	0*1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Note 1. Depends of the read timing after the CPU starts operation.

Table 11.5 PLL0CLKS Register Contents

Bit Position	Bit Name	Function
31 to 3	—	Reserved These bits are always read as 0.
2	CLKACT	PLL0 Clock Source State 0: PLL0 clock source is inactive. 1: PLL0 clock source is active.
1	CLKSTAB	PLL0 Clock Stable State 0: PLL0 is unstable. 1: PLL0 is stable.
0	CLKEN	PLL0 Clock Operating State 0: PLL0 is stopped. 1: PLL0 is operating.

11.4.3 PLL0CLKC1 — PLL0 Control Register 1

This register controls the operation of the spread spectrum clock generator (SSCG) for PLL0.

This register can be set when PLL0 is running.

This register is protected by the PROT1PHCMD register.

This register is initialized by either an internal or external reset.

Access: This register can be read/written in 32-bit units.

Address: FFF8 8200$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	SSMODE1	—	SELMFREQ[4:0]					SELMPERCENT[2:0]		
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 11.6 PLL0CLKC1 Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 10	—	Reserved These bits are always read as 0. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
9	SSMODE1	SSCG Control 0: Modulation disabled 1: Modulation enabled
8	—	Reserved This bit is always read as 0. When writing to this bit, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
7 to 3	SELMFREQ [4:0]	SSCG Cycle Setting The modulation cycle of SSCG can be selected by setting these bits. Selectable settings are as follows: 10000_B : 80.65 kHz 10001_B : 75.76 kHz 10010_B : 69.44 kHz 10011_B : 65.79 kHz 10100_B : 62.50 kHz 10101_B : 59.52 kHz 10110_B : 58.14 kHz 10111_B : 50.00 kHz 11000_B : 41.67 kHz 11001_B : 39.68 kHz 11010_B : 37.31 kHz 11011_B : 33.33 kHz 11100_B : 30.12 kHz 11101_B : 25.00 kHz 11110_B : 20.00 kHz When SSMODE1 is set to 1, do not set a value other than the above.

Table 11.6 PLL0CLKC1 Register Contents (2/2)

Bit Position	Bit Name	Function
2 to 0	SELMPERCENT [2:0]	SSCG Modulation Range Setting The modulation range of SSCG can be selected by setting these bits. Selectable settings are as follows: $1\ 0\ 0_B$: -5.0% Do not set any other value while SSMODE1 is set to 1.

When changing the modulation depth of PLL0, turn off the SSCG and then wait for at least 1.6 ms.

11.4.4 CKSC0CTL — Clock 0 Selection Control Register

This register is used for selection of a clock source for the frequency divider 0A.

This register is protected by the PROT1PHCMD register.

This register is initialized by either an internal or external reset.

Access: This register can be read/written in 32-bit units.

Address: FFF8 9000_H

Value after reset: 0000 0020_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	CKSC[5:0]					
Value after reset	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Table 11.7 CKSC0CTL Register Contents

Bit Position	Bit Name	Function
31 to 6	—	Reserved These bits are always read as 0. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
5 to 0	CKSC[5:0]	These bits select a clock source for the frequency divider 0A. 100000_B: Selects the PLL1 clock. 100011_B: Selects the PLL0 clock. Setting other than the above is prohibited.

11.4.5 CKSC0ACT — Clock 0 Selection Active Register

This register indicates the state of a clock source for the frequency divider 0A.

This register is initialized by either an internal or external reset.

Access: This register can be read in 32-bit units.

Address: FFF8 9008_H

Value after reset: $0000\ 0020_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	CLKACT[5:0]					
Value after reset	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 11.8 CKSC0ACT Register Contents

Bit Position	Bit Name	Function
31 to 6	—	Reserved These bits are always read as 0.
5 to 0	CLKACT[5:0]	These bits indicate that switching of a clock source for divider 0A is completed when their values are the same as those specified in the CKSC[5:0] bits in the CKSC0CTL register.

11.4.6 CLKD0DIV — Clock 0 Division Register

This register specifies a frequency division ratio of divider 0A.

This register is protected by the PROT1PHCMD register.

This register is initialized by either an internal or external reset.

Access: This register can be read/written in 32-bit units.

Address: FFF8 8800_H

Value after reset: 0000 0004_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	CLKD0DIV[2:0]		
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Table 11.9 CLKD0DIV Register Contents

Bit Position	Bit Name	Function
31 to 3	—	Reserved These bits are always read as 0. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
2 to 0	CLKD0DIV[2:0]	These bits select a frequency division ratio of divider 0A. 001_B: No division 010_B: Divided by 2 100_B: Divided by 4 Setting other than the above is prohibited.

11.4.7 CLKD0STAT — Clock 0 Division Status Register

This register indicates a clock state of the frequency divider 0A.

This register is initialized by either an internal or external reset.

Access: This register can be read in 32-bit units.

Address: FFF8 8804$_H$

Value after reset: 0000 0001$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLKD0 SYNC
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 11.10 CLKD0STAT Register Contents

Bit Position	Bit Name	Function
31 to 1	—	Reserved These bits are always read as 0.
0	CLKD0SYNC	Divider Synchronization Status 0: Divider 0A is switching to the division ratio specified in CLKD0DIV. 1: Divider 0A is running at the division ratio specified in CLKD0DIV.

11.4.8 CKSC1CTL — Clock 1 Selection Control Register

This register is used for selection of an output clock for the frequency divider 1A.

This register is protected by the PROT1PHCMD register.

This register is initialized by either an internal or external reset.

Access: This register can be read/written in 32-bit units.

Address: FFF8 9040_H

Value after reset: 0000 0021_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	CKSC[5:0]					
Value after reset	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Table 11.11 CKSC1CTL Register Contents

Bit Position	Bit Name	Function
31 to 6	—	Reserved These bits are always read as 0. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
5 to 0	CKSC[5:0]	These bits select an output clock for the frequency divider 1A. 100001_B: Selects 1/4 clock of PLL1. 100010_B: Selects 1/2 clock of PLL1. 100100_B: Selects 1/1 clock of PLL1. Setting other than the above is prohibited.

11.4.9 CKSC1ACT — Clock 1 Selection Active Register

This register indicates a clock state of the frequency divider 1A.

This register is initialized by either an internal or external reset.

Access: This register can be read in 32-bit units.

Address: FFF8 9048$_H$

Value after reset: 0000 0021$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	CLKACT[5:0]					
Value after reset	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 11.12 CKSC1ACT Register Contents

Bit Position	Bit Name	Function
31 to 6	—	Reserved These bits are always read as 0.
5 to 0	CLKACT[5:0]	These bits indicate that switching of an output clock for the frequency divider 1A is completed when their values are the same as those specified in the CKSC[5:0] bits in the CKSC1CTL register. 100001_B: 1/4 of PLL1 100010_B: 1/2 of PLL1 100100_B: 1/1 of PLL1

11.4.10 PROT1PHCMD — Protect 1 Command Register

PROT1PHCMD is a protection command register to start the protection unlocking sequence necessary for access to write-protected registers.

This register is initialized by either an internal or external reset.

Access: This register can be read/written in 32-bit units.

Address: FFF8 $B000_H$

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	PCMD[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 11.13 PROT1PHCMD Register Contents

Bit Position	Bit Name	Function
31 to 8	—	Reserved These bits are always read as 0.
7 to 0	PCMD[7:0]	Write protection command register

This register is used in protecting against inadvertent access to write-protected registers, i.e. registers to which writing raises the possibility of serious effects on application systems, such as programs crashing and the like.

Writing to the write-protected registers is only possible with the protection unlock sequence described below.

1. Write the fixed value (0000 $00A5_H$) to register PROT1PHCMD.
2. Write the new setting to the write-protected register. Write the value after a reset to the reserved bits.
3. Write the bitwise inverse of the setting value to the same register as in step 2. Write the inverse of the value after a reset to the reserved bits.
4. Write the new setting to the same register as in step 2. Write the value after a reset to the reserved bits.

A value can only be written to a write-protected register by following the procedure above.

Protection is not unlocked if the procedure is not followed correctly, the setting is not written to the write-protected register, and the PROT1PS.PROTERR bit is set to 1. (Although this is not a requirement, you can confirm if the values set in the targeted register was correctly written by checking that the setting of the PROT1PS.PROTERR bit is 0 after step 4.)

In case of failure to follow the sequence, repeat the procedure from the beginning.

In case of writing to another register during steps 1 to 4 of the sequence, the protection function operates as follows. Operation is also as described when any interrupt is accepted during the sequence and the interrupt handling includes access to another register.

- When the writing is to another register in the same module*1, writing to the protected register fails and the PROT1PS.PROTERR bit is set to 1.
- When writing is to a register in another module, writing to the protected register will successfully proceed to completion, if the remainder of the procedure is followed correctly.

Reading of another register while the sequence is in progress does not lead to failure.

Note 1. PROT1PHCMD applies to the registers allocated to the addresses in the range from FFF8 8004_H to FFF8 $B004_H$.
For the register names, register symbols, and module names, see **Appendix, List of Registers**.

11.4.11 PROT1PS — Protect 1 Status Register

PROT1PS is a status register of the protection unlocking sequence. This register indicates whether an error occurs or not in access to write-protected registers.

This register is initialized by either an internal or external reset.

Access: This register can be read/written in 32-bit units.

Address: FFF8 B004$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PROT ERR
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 11.14 PROT1PS Register Contents

Bit Position	Bit Name	Function
31 to 1	—	Reserved These bits are always read as 0.
0	PROTERR	Protection Error Flag 0: Protection error does not occur. 1: Protection error occurs.

Operating conditions of the PROTERR bit

[Setting Condition]

Access to a write-protected register without executing the step of the protection unlocking sequence that involves the PROT1PHCMD register

[Clearing Condition]

Writing of 0000 00A5$_H$ to the PROT1PHCMD register (step 1 in the protection unlocking sequence)

11.5 Operation

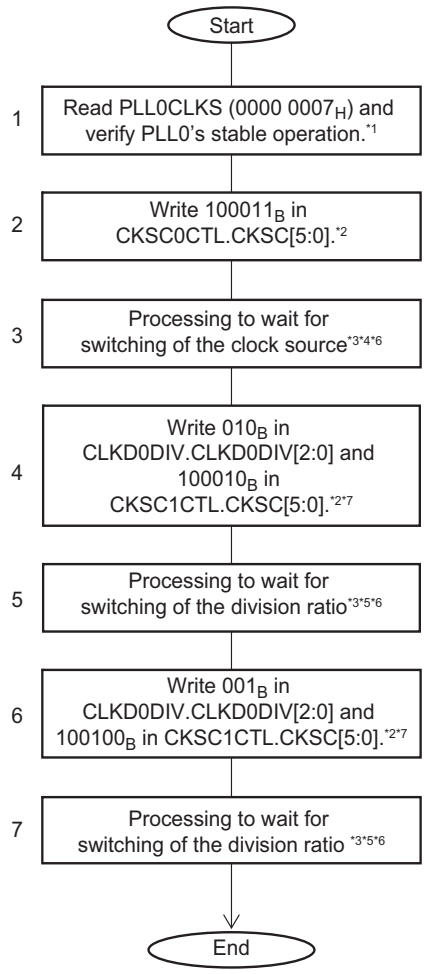
11.5.1 Operation When the Divide Function Is Used

Follow the procedure given below to reduce current fluctuations produced by switching the clock signals during startup (also see **Figure 11.3, Example of Sequence for Shifting the Clock Gear Up**).

1. When a user program is running after release from the reset state, PLL0 or PLL1 will be oscillating and PLL1 will be in use as the internal operating clock. Also, the division ratio of divider 0A is set to 1/4.
   Read the PLL0CLKS register and verify that its value is 07_H, indicating that PLL0 operation is stable.
2. Write 23_H to the CKSC[5:0] bits in CKSC0CTL to select PLL0 as the clock source.
3. When switching the clock source, wait at least 105 or 140 cycles of the CPU clock in the respective cases of products with CPUs having maximum operating frequencies of 240 MHz or 320 MHz . After that, read CKSC0ACT and verify that the value of the CKSC0ACT.CLKACT[5:0] bits is 23_H.
4. Write 010_B to the CLKD0DIV[2:0] bits in CLKD0DIV to select 1/2 as the division ratio for the divider.
5. For switching of the division ratio, wait at least, 105 or 140 cycles of the CPU clock in the respective cases of products with CPUs having maximum operating frequencies of 240 MHz or 320 MHz . Then read CLKD0STAT and verify that the value of the CLKD0SYNC is 1.
6. Write 001_B to the CLKD0DIV[2:0] bits in CLKD0DIV to select 1/1 as the division ratio for the divider.
7. For switching of the division ratio, wait at least 105 or 140 cycles of the CPU clock in the respective cases of products with CPUs having maximum operating frequencies of 240 MHz or 320 MHz . Then read CLKD0STAT and verify that the value of the CLKD0SYNC is 1.

For switching division ratio of the divider 1A on the PLL1, follow the procedures of 4 to 7 above.

Figure 11.3 shows an example of the procedure for switching the clocks (and shifting the clock gear up for both clocks). **Table 11.15** lists the clock frequencies in each step of the process.



Note 1. After release from the reset state, verify that PLL0 operation is stable before executing the sequence to shift the clock gear up. PLL0CLKS = 07_H indicates that PLL0 is stable.

Note 2. The CKSC0CTL and CLKD0DIV registers are protected by the PROT1PHCMD register. Execute the protection unlocking sequence (see the description of the protect 1 command register) when setting these registers.

Note 3. This processing is required because actual switching of the clock source or division ratio from the previous register settings takes time. The processing must produce periods of waiting of at least 105 or 140 cycles of the CPU clock in the respective cases of products with CPUs having maximum operating frequencies of 240 or 320 MHz . The SNOOZE instruction can be used for this purpose; see the RH850G3M User's Manual: Software for details.

Note 4. Verify that the value of the CLKACT[5:0] bits in CKSC0ACT is the same as that specified by the CKSC[5:0] bits in CKSC0CTL after switching the source clock for frequency divider 0A. Similarly, verify that the value of the CLKACT[5:0] bits in CKSC1ACT is the same as that specified by the CKSC[5:0] bits in CKSC1CTL after switching the source clock for frequency divider 1A.

Note 5. Verify that the CLKD0SYNC bit in CLKD0STAT is 1 after changing the division ratio of divider 0A.

Note 6. Switching of clock sources or division ratios produces transient voltage fluctuations which last for a certain time, so measures such as processing to add more cycles of waiting may be required. Whether or not this is the case will depend on evaluation of the actual environment where the LSI chip is used.

Note 7. In the procedure described above, the division ratios of both PLL0 and PLL1 are set at the same time. When minimizing the transient voltage fluctuations during this procedure is crucial, separate the register settings and subsequent periods of waiting for PLL0 and PLL1 to achieve this (which also prolongs the procedure).

Figure 11.3 Example of Sequence for Shifting the Clock Gear Up

Table 11.15 Frequencies in the Process of Shifting the Clock Gear Up

	CLK_CPU					
	Products with 320-MHz (max.) CPUs	Products with 240-MHz (max.) CPUs	CLK_HSB	CLK_LSB	CLKC_HSB	CLKC_LSB
Before Step 1	80	60	20	10	20	10
Between Step 1 and Step 2	80	60	20	10	20	10
Between Step 2 and Step 3	160	120	40	20	40	20
After Step 3	320	240	80	40	80	40

11.6 Notes

Avoid using commands that employ the computing unit, such as FPU operations, while the division ratio is being changed. Increased current fluctuation may make the operation unstable.

11.6.1 Board Design Notes

As shown in **Figure 11.4**, do not cross any other signal lines over the signal lines to the EXTAL and XTAL pins. Induction may inhibit proper oscillation.

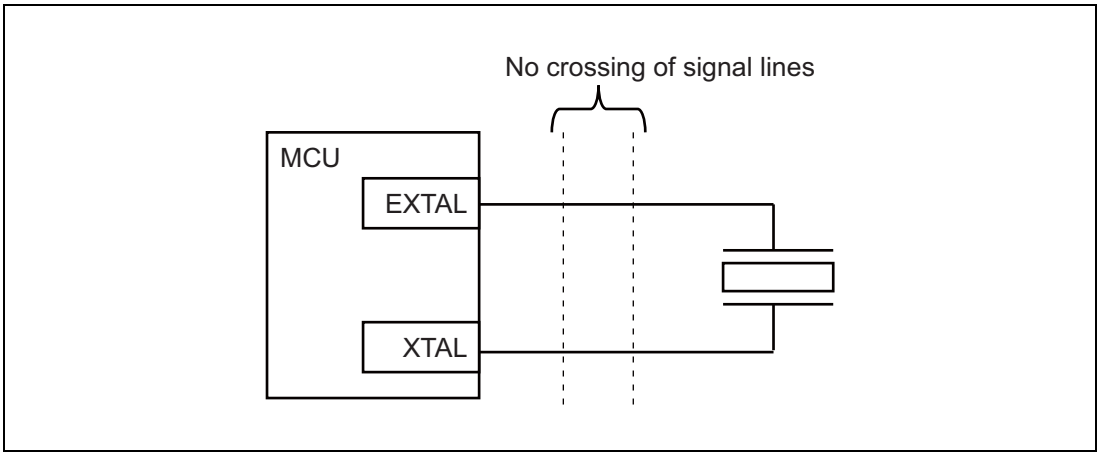


Figure 11.4 Board Design Notes

11.7 ASIC Clock

11.7.1 Features

- ASIC Clock provides a function to select a clock output from the CK pin.
- The frequency-divided clock*2 set by the 1-MHz clock*1 or the baud rate generator (BRG) can be selected through the register settings.
- After release from power-off standby, the 1-MHz clock is output from the CK pin.
- When stop or run of a clock is selected, rewriting to a register should be done in a specific sequence so that the register is not incorrectly rewritten due to program malfunction.

CAUTIONS

1. 1-MHz clock: 1/20 clock of the 20-MHz main OSC

2. BGR (Baud Rate Generator): 1/N clock of the unmodulated low speed peripheral clock (CLKC_LSB)

11.7.2 Configuration

Figure 11.5 shows the diagram of ASIC clock output configuration.

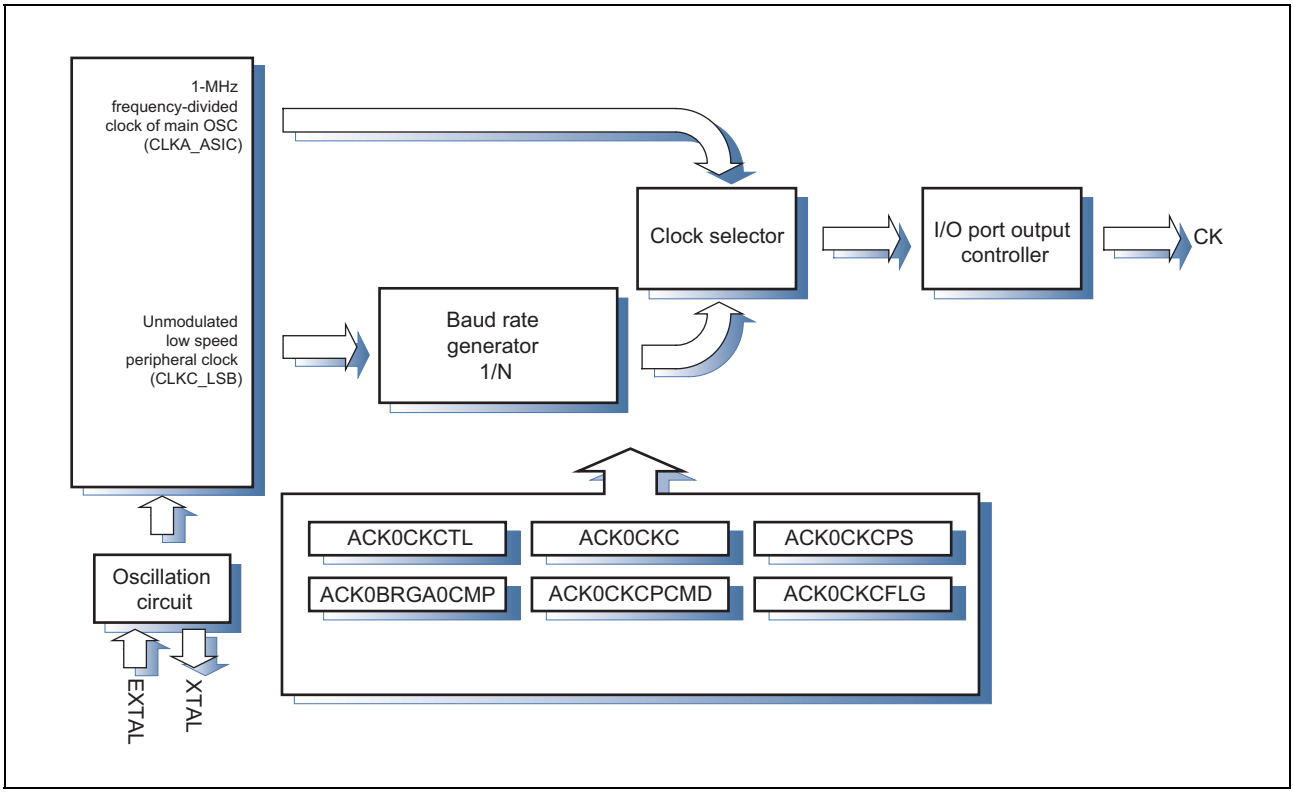


Figure 11.5 Block Diagram of ASIC Clock

11.7.3 Register Specifications

The following shows the specifications of registers.

Table 11.16 Pin Specifications

Register Name	Abbreviation	R/W	Value after Reset	Address	Access Size
Clock control register	ACK0CKC	R/W	01_H	FFC0 4000_H	8
CKC flag register	ACK0CKCFLG	R	01_H	FFC0 4004_H	8
Clock select register	ACK0CKCTL	R/W	00_H	FFC0 4008_H	8
BRGA0 compare register	ACK0BRGA0CMP	R/W	00_H	FFC0 $400C_H$	8
CKC protect command register	ACK0CKCPCMD	W	—	FFC0 4100_H	8
CKC protect status register	ACK0CKCPS	R	00_H	FFC0 4104_H	8

11.7.3.1 ACK0CKC — Clock Control Register

This is an 8-bit register to control the system clock frequency. Therefore, this register can be only written in combination of specific instruction sequences so that it is not incorrectly rewritten due to program malfunction.

Protection of this register can be canceled by writing to the command register (ACK0CKCPCMD).

When a specific register is read out, however, a specific instruction sequence is not required.

Access: This register can be read/written in 8-bit units.

Address: FFC0 4000_H

Value after reset: 01_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	DCLK0
Value after reset	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R/W

Table 11.17 ACK0CKC Register Contents

Bit Position	Bit Name	Description
7 to 1	—	Reserved These bits are always read as 0. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
0	DCLK0	Clock Output Control 0: Clock output inhibit mode Output of a clock from the CK pin is prohibited (fixed to low level). Further reduction of power consumption and suppression of radiation noise from the CK pin can be achieved because CK operation is completely stopped. 1: Clock output The clock selected by the ACK0CKCTL is output from the CK pin.

Note 1. Up to three cycles of the clock signal output from the the CK pin (the clock signal selected by ACK0CKCTL) are required to stop clock output to the CK pin after writing 0 to the DCLK0 bit. The clock output stop state can be checked by reading the DCLKF0 bit in the ACK0CKCFLG register.

11.7.3.2 ACK0CKCFLG — CKC Flag Register

The ACK0CKCFLG register indicates the status of clock output control.

Access: This register can be read in 8-bit units.

Address: FFC0 4004_H

Value after reset: 01_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	DCLKF0
Value after reset	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R

Table 11.18 ACK0CKCFLG Register Contents

Bit Position	Bit Name	Description
7 to 1	—	Reserved These bits are always read as 0.
0	DCLKF0	This bit indicates the status of clock output control. 0: Output from the CK pin is stopped. 1: The clock selected by the ACK0CKCTL is output from the CK pin.

11.7.3.3 ACK0CKCTL — Clock Select Register

The ACK0CKCTL is an 8-bit register to select a clock output from the CK pin.

This register can be written in 8-bit units when the DCLKF0 bit in the CKC flag register is 0.

The ACK0CKCTL register is always readable also.

For the DCLKF0 bit, see **Section 11.7.3.2, ACK0CKCFLG — CKC Flag Register**.

Access: This register can be read/written in 8-bit units.

Address: FFC0 4008_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CKSL[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 11.19 ACK0CKCTL Register Contents

Bit Position	Bit Name	Description
7 to 2	—	Reserved These bits are always read as 0. When writing, always write 0.
1, 0	CKSL[1:0]	These bits select a clock output from the CK pin. 00: 1-MHz clock (1/20 clock of the 20-MHz main OSC) 01: BRG (Baud Rate Generator) Setting other than the above is prohibited.

Note 1. Writing to the ACK0CKCTL register (selection of a clock output to the CK pin) can only be performed when the clock output to the CK pin is stopped (the DCLKF0 bit in the ACK0CKCFLG register is 0).

11.7.3.4 ACK0BRGA0CMP — BRGA0 Compare Register

The ACK0BRGA0CMP is a register to store a value to be compared with the baud rate counter.

Access: This register can be read/written in 8-bit units.

Address: FFC0 400C$_H$

Value after reset: 00$_H$

Bit	7	6	5	4	3	2	1	0
	—	—	—	BRGA0CMP[4:0]				
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Table 11.20 ACK0BRGA0CMP Register Contents

Bit Position	Bit Name	Description
7 to 5	—	Reserved These bits are always read as 0. When writing, always write 0.
4 to 0	BRGA0CMP [4:0]	These bits set a compare value. For the settings, see **Table 11.22, Unmodulated Low Speed Peripheral Clock Frequency (value set in the ACK0BRGA0CMP register and frequency to be output)**.

Note 1. Writing to the ACK0BRGA0CMP register (selection of a value to be compared with the baud rate counter) can only be performed when the clock output to the CK pin is stopped (the DCLKF0 bit in the CKCFLG register is 0).

11.7.3.5 ACK0CKCPCMD — CKC Protect Command Register

The ACK0CKCPCMD register is used to protect against any writing operation to a register that might have a material effect on the application systems. This ensures the systems are not incorrectly stopped due to program malfunction.

The value of the ACK0CKC register can only be rewritten by a sequence specified in advance, and an unauthorized write operation is eliminated.

Access: This register can be written in 8-bit units. The read value is always undefined.

Address: FFC0 4100_H

Value after reset: Undefined

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—
Value after reset	—	—	—	—	—	—	—	—
R/W	W	W	W	W	W	W	W	W

Writing to the ACK0CKC register is only possible with the protection unlock sequence described below.

Procedure

1. Write the fixed value ($A5_H$) to register ACK0CKCPCMD.
2. Write the new setting to register ACK0CKC. Write the value after a reset to the reserved bits.
3. Write the bitwise inverse of the setting value to register ACK0CKC. Write the inverse of the value after a reset to the reserved bits.
4. Write the new setting to register ACK0CKC. Write the value after a reset to the reserved bits.

A value can only be written to register ACK0CKC by following the procedure above.

Protection is not unlocked if the procedure is not followed correctly, the setting is not written to the ACK0CKC register, and the ACK0CKCPS.CKCPRERR bit is set to 1. (Although this is not a requirement, you can confirm if the values set in the targeted register was correctly written by checking that the setting of the ACK0CKCPS.CKCPRERR bit is 0 after step 4.)

In case of failure to follow the sequence, repeat the procedure from the beginning.

In case of writing to another register during steps 1 to 4 of the sequence, the protection function operates as follows. Operation is also as described when any interrupt is accepted during the sequence and the interrupt handling includes access to another register.

- When the writing is to another register in the same module*1, writing to the protected register fails and the ACK0CKCPS.CKCPRERR bit is set to 1.
- When writing is to a register in another module, writing to the protected register will successfully proceed to completion, if the remainder of the procedure is followed correctly.

Reading of another register while the sequence is in progress does not lead to failure.

Note 1. "Another register in the same module" indicates a register under the same module name as that containing the write-protected register. For the module names of the registers, see **Appendix, List of Registers**.

11.7.3.6 ACK0CKCPS — CKC Protect Status Register

The ACK0CKCPS is a register to which the CKCPRERR flag indicating a protection error occurrence is allocated.

If the ACK0CKC register is not written to in the correct sequence, including the command register settings, the write operation is not performed on the target register. A protection error then occurs and the status flag (CKCPRERR) is set to 1.

The ACK0CKCPS register is cleared at each start of the specific sequence.

Access: This register can be read in 8-bit units.

Address: FFC0 4104_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	CKCPR ERR
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 11.21 ACK0CKCPS Register Contents

Bit Position	Bit Name	Description
7 to 1	—	Reserved These bits are always read as 0.
0	CKCPRERR	Protection Error Detection 0: Protection error has not occurred. 1: Protection error has occurred.

11.7.4 Usage

11.7.4.1 CK Pin Output Function

A clock source output from the CK pin can be selected and provided in this function.

1-MHz clock is output from the CK pin during the reset period immediately after release from power-off standby. A clock source output from the CK pin can be selected among the 1-MHz clock and the frequency-divided clocks set by the BRG (Baud Rate Generator) through switching of the ACK0CKCTL register.

- Selectable clock source
  1-MHz clock (1/20 clock of the 20-MHz main OSC) (Initial value)
  BRG (Baud Rate Generator)

11.7.4.2 CK Output Baud Rate Generator Circuit (BRGA)

In the baud rate generator circuit, the unmodulated low speed peripheral clock (CLKC_LSB) is divided and output from the CK pin.

The BRGA0 inverts the CK output in response to a match in comparison between the specified value of the BRGA0CMP[4:0] bit in the ACK0BRGA0CMP register and the value of the baud rate counter.

The CK output is fixed to low when the DCLK0-bit in the ACK0CKC register is set to 0.

Table 11.22 lists the frequencies of unmodulated low speed peripheral clocks that can be set.

Table 11.22 Unmodulated Low Speed Peripheral Clock Frequency (value set in the ACK0BRGA0CMP register and frequency to be output)

BRGA0CMP4	BRGA0CMP3	BRGA0CMP2	BRGA0CMP1	BRGA0CMP0	Output Frequency [MHz]
0	0	0	0	0	1.25
1	0	1	0	0	2.00
1	0	1	0	1	1.90
1	0	1	1	0	1.82
1	0	1	1	1	1.74
1	1	0	0	0	1.67
1	1	0	0	1	1.60
1	1	0	1	0	1.54
1	1	0	1	1	1.48
1	1	1	0	0	1.43
1	1	1	0	1	1.38
1	1	1	1	0	1.33
1	1	1	1	1	1.29

Note 1. Setting other than the above is prohibited.

11.7.4.3 Control Register Rewrite Protect

As shown in **Figure 11.6**, the ACK0CKC register to control the output clock from the CK pin can only be rewritten in a specific sequence due to this protect function. Switching of a clock source is prohibited while a clock is running. The ACK0CKCTL register can only be rewritten when the DCLKF0 bit in the CKC flag register is set to 0.

In addition, rewriting of the compare values is prohibited while the baud rate counter is running. The ACK0BRGA0CMP register can only be rewritten when the DCLKF0 bit in the CKC flag register is set to 0.

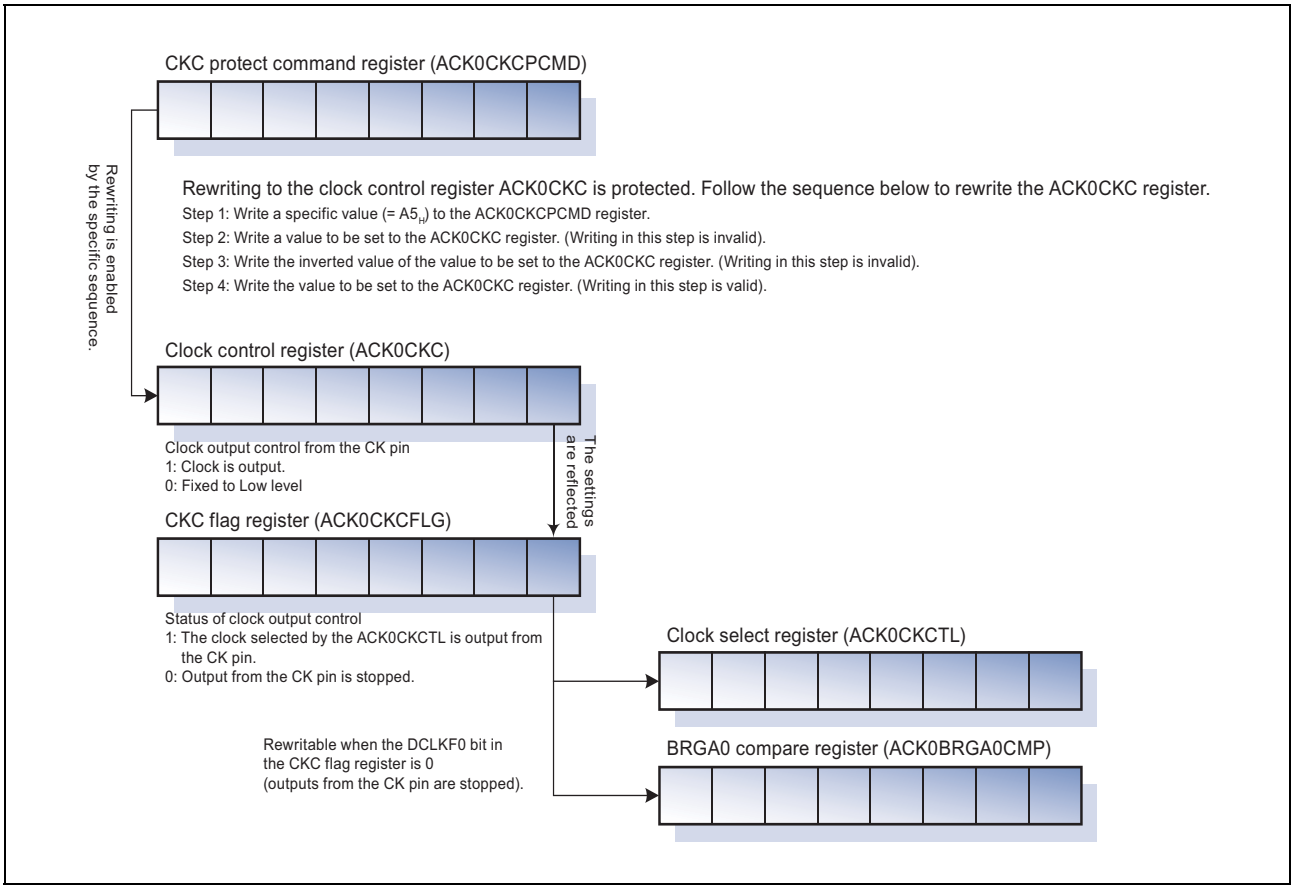


Figure 11.6 Rewriting Protect

11.7.4.4 CK Output Switching Procedure

Follow the procedure below to switch the output clock from the CK pin.

1. Stop an output from the CK pin.

   Rewrite the DCLK0 bit in the clock control register (ACK0CKC) to stop an output from the CK pin.

- Write $A5_H$ to the ACK0CKCPCMD register.
- Write 00_H to the ACK0CKC register.
- Write FF_H to the ACK0CKC register.
- Write 00_H to the ACK0CKC register.

  Read the DCLKF0 bit in the CKC flag register (ACK0CKCFLG) to verify that the output from the CK pin has been stopped.

2. Select a clock source

   Rewrite the CKSL[1:0] bits in the clock selection register (ACK0CKCTL) to select a clock source.
   Also rewrite the BRGA0CMP[4:0] bits in the BRGA0 compare register (ACK0BRGA0CMP) to select a clock cycle for the baud rate generator.

3. Restart an output from the CK pin

   Rewrite the DCLK0 bit in the clock control register (ACK0CKC) to restart an output from the CK pin.

- Write $A5_H$ to the ACK0CKCPCMD register.
- Write 01_H to the ACK0CKC register.
- Write FE_H to the ACK0CKC register.
- Write 01_H to the ACK0CKC register.

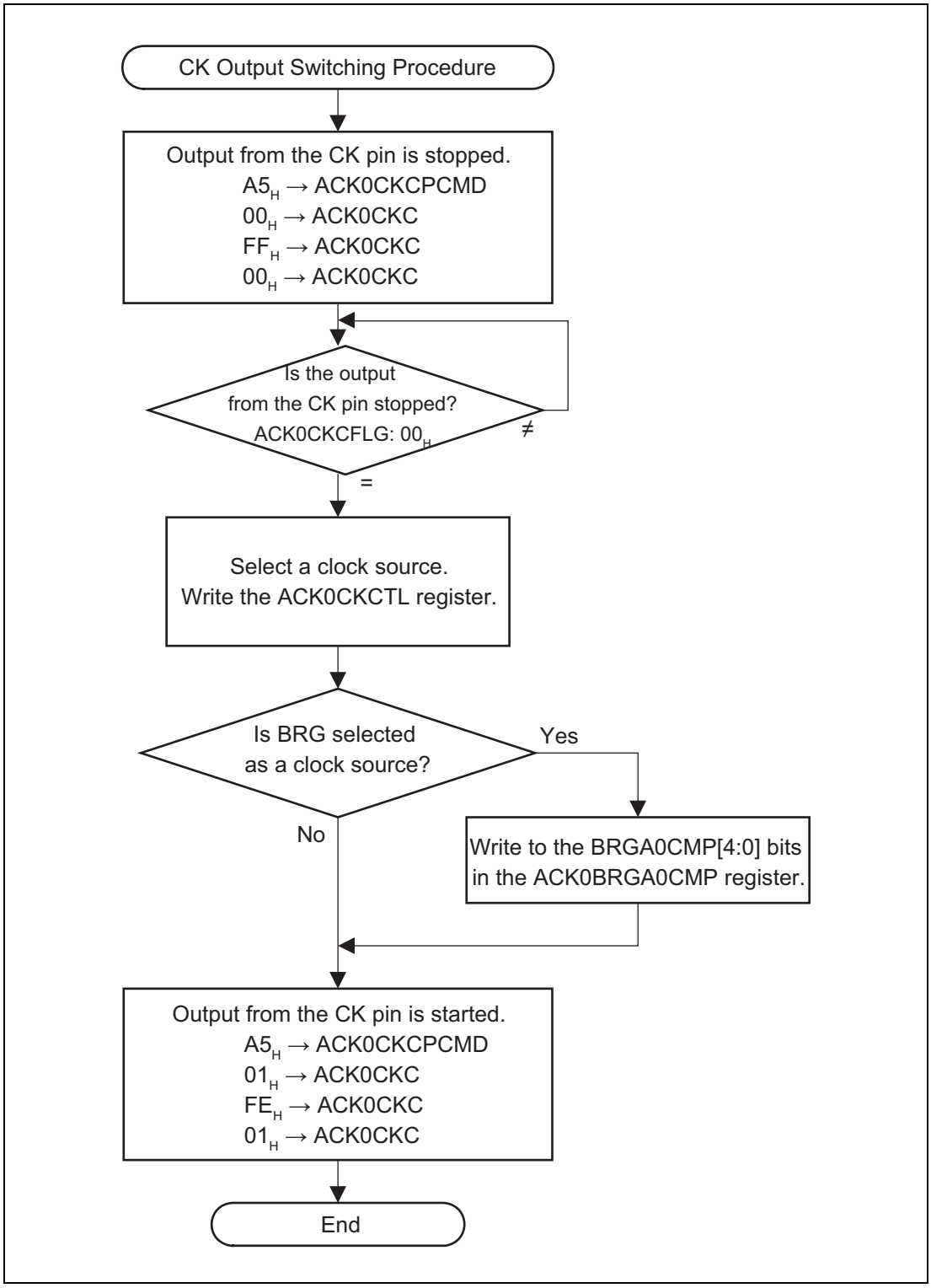


Figure 11.7 CK Output Switching Procedure

Section 12 Standby Controller

This product supports various power-down modes: two modes of power off standby mode and HALT mode, and module standby function that can stop peripheral modules. The power consumption of the LSI can be reduced by selecting a suitable mode for the application.

12.1 Features

12.1.1 Types of Power-Down Mode

Power-down modes are comprised of the modes and function as follows:

- Power off standby mode
  Only RAM retention is possible in this mode. Low leakage is realized by turning off the power outside of the LSI. The oscillation also stops.
  The transition occurs by asserting the $\overline{\text{RES}}$ pin and turning off the power (EVCC).
  The CPU in this state transits to the external reset state by a rising edge of the turned off power and securing the oscillation stabilization time.
- HALT mode
  When the HALT instruction is executed, the CPU transits to HALT mode and stops instruction execution.
  Each CPU can be controlled individually.
  The CPU returns from this state by the occurrence of a reset input, interrupt, or exception.
- Module standby
  This function stops the clock for peripheral macros to reduce the power consumption in accordance with register settings.
  Since peripheral modules are in operation after release from the reset state, it is recommended that the modules not to be used be set to the module stop state before clock gear-up.
  Modules which have been set in the module stop state should not be activated again. In such a case, proper operation cannot be guaranteed.

12.2 Operation

12.2.1 Power Off Standby Mode

The CPU transits to power-off standby mode when EVCC power is set to 0V after driving the $\overline{\text{RES}}$ pin to the low level. In power-off standby mode, leakage current can be reduced because the oscillator is stopped and the core power is turned off. For mid-range and low-end products with the EPT, the EPT controls VDD turn-off.

In power-off standby mode, the data in the RAM can be retained as long as SYSVCC is kept within the specified voltage range.
For the target RAMs, see **Section 35, RAM**.

For recovery from power-off standby mode, restore EVCC and VCC to the specified voltage to allow the CPU to return to the external reset state. Then negate the $\overline{\text{RES}}$ pin after securing the oscillation stabilization time to allow the CPU to return to the normal state.

Refer to **Section 37, Electrical Characteristics**, for the power-off sequences.

12.2.2 Module Standby Function

(1) Transition to the module standby state

Setting each bit in the module standby register to 1 can stop clock supply to each corresponding on-chip peripheral module.

It is recommended that the module standby state be set for power consumption reduction before clock gear-up (while the module is disabled).

The module standby register is protected from unauthorized writing due to incorrect program operation and the like.

For details, see **Section 12.3.6, MSRPCMD — MSR Protection Command Register**.

(2) Release from the module standby state

Set the LSI to the internal reset state for release from the module standby state.

Proper operation cannot be guaranteed when the module standby state which has been already set is released by writing 0 to the module standby register.

12.3 Register List

12.3.1 List of Registers

All registers can be accessed by CPU.

Table 12.1 Register Configuration

Address	Register name	Description	Access Width	Value after Reset	Access protection
FFF8 2C0C	EPTCNT	EPT control register	32	0000 0000_H	PROT0PHCMD
FFF8 3000	PROT0PHCMD	Protection command register	32	0000 0000_H	
FFF8 3004	PROT0PS	Protection command status register	32	0000 0000_H	
FFC0 5000	MSRTSG	Module standby register - TSG2	8	00_H	MSRPCMD
FFC0 5004	MSRTAPA	Module standby register - TAPA	8	00_H	MSRPCMD
FFC0 5008	MSROSTM	Module standby register - OSTM	8	00_H	MSRPCMD
FFC0 500C	MSRWDTA	Module standby register - WDTA	8	00_H	MSRPCMD
FFC0 5010	MSRPIC	Module standby register - PIC	8	00_H	MSRPCMD
FFC0 5014	MSRRCAN	Module standby register - RS-CAN	8	00_H	MSRPCMD
FFC0 5018	MSRFRAY	Module standby register - FlexRay	8	00_H	MSRPCMD
FFC0 501C	MSRRLIN	Module standby register - RLIN2	8	00_H	MSRPCMD
FFC0 5020	MSRSCI	Module standby register - SCI3	8	00_H	MSRPCMD
FFC0 5024	MSRCSIH	Module standby register - CSIH	8	00_H	MSRPCMD
FFC0 5028	MSRSAD	Module standby register - SAR AD	8	00_H	MSRPCMD
FFC0 502C	MSRDAD	Module standby register - ΔΣAD	8	00_H	MSRPCMD
FFC0 5030	MSRATU	Module standby register - ATU-IV	8	00_H	MSRPCMD
FFC0 5034	MSRAPA	Module standby register - APA	8	00_H	MSRPCMD
FFC0 5038	MSRDFE	Module standby register - DFE	8	00_H	MSRPCMD
FFC0 503C	MSRRHSB	Module standby register - RHSB	8	00_H	MSRPCMD
FFC0 5070	MSRPCMD	MSR protection command register	8	—	
FFC0 5074	MSRPS	MSR protection status register	8	00_H	

12.3.2 EPTCNT — EPT Control Register

This register enables or disables the EPT.
This register is protected by the PROT0PHCMD register.
This register can only be reset by an external reset signal.

Access: This register can be read/written in 32-bit units.
Address: FFF8 2C0C
Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	EPT CNT
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 12.2 EPTCNT Register Contents

Bit Position	Bit Name	Function
31 to 1	—	Reserved The read value is always 0. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
0	EPTCNT	0: EPT is enabled. 1: EPT is disabled.

When the EPT is not used, the EPT control can be powered down by setting 1 in the register above, resulting in power consumption reduction.

12.3.3 PROT0PHCMD — Protection Command Register

This register is used as a special sequence register.

This register can be reset by either an internal or external reset.

Access: This register can be read/written in 32-bit units.

Address: FFF8 3000

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	PCMD[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 12.3 PROT0PHCMD Register Contents

Bit Position	Bit Name	Function
31 to 8	—	Reserved The read value is always 0. When writing, write 0.
7 to 0	PCMD[7:0]	Writing protection command register

Writing to the write-protected registers is only possible with the protection unlock sequence described below.

1. Write the fixed value (0000 $00A5_H$) to register PROT0PHCMD.
2. Write the new setting to the write-protected register. Write the value after a reset to the reserved bits.
3. Write the bitwise inverse of the setting value to the same register as in step 2. Write the inverse of the value after a reset to the reserved bits.
4. Write the new setting to the same register as in step 2. Write the value after a reset to the reserved bits.

A value can only be written to a write-protected register by following the procedure above.

Protection is not unlocked if the procedure is not followed correctly, the setting is not written to the write-protected register, and the PROT0PS.PROTERR bit is set to 1. (Although this is not a requirement, but you can confirm if the values set to the targeted register was correctly written by checking that the setting of the PROT0PS.PROTERR bit is 0 after step 4).

In case of failure to follow the sequence, repeat the procedure from the beginning.

In case of writing to another register during steps 1 to 4 of the sequence, the protection function operates as follows:

Operation is also as described when any interrupt is accepted during the sequence and the interrupt handling includes access to another register.

- When the writing is to another register in the same module*1, writing to the protected register fails and the PROT0PS.PROTERR bit is set to 1.
- When writing is to a register in another module, writing to the protected register will successfully proceed to completion, if the remainder of the procedure is followed correctly.

Reading of another register while the sequence in progress does not lead to failure.

Note 1. PROT0PHCMD applies to the registers allocated to the addresses in the range from FFF8 2410_H to FFF8 3004_H.
For the register names, register symbols, and module names, see Appendix, List of Registers. see **Appendix, List of Registers**.

12.3.4 PROT0PS — Protection Command Status Register

This register is used as a status register of special sequence registers. This register can be reset by either an internal or an external reset signal.

Access: This register can be read in 32-bit units.

Address: FFF8 3004

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PROT ERR
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 12.4 PROT0PS Register Contents

Bit Position	Bit Name	Function
31 to 1	—	Reserved The read value is always 0.
0	PROTERR	Protection error flag

12.3.5 Module Standby Registers

The module standby registers can be read/written in 8-bit units and control each module operation in power-down mode.

12.3.5.1 MSRTSG — Module Standby Register – TSG2

Access: This register can be read/written in 8-bit units.

Address: FFC0 5000

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	MS_TSG21	MS_TSG20
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 12.5 MSRTSG Register Contents

Bit Position	Bit Name	Function
7 to 2	—	Reserved The read value is always 0. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
1	MS_TSG21	Setting this bit to 1 stops clock supply to TSG21. 0: TSG21 is operating. 1: Clock supply to TSG21 is stopped.
0	MS_TSG20	Setting this bit to 1 stops clock supply to TSG20. 0: TSG20 is operating. 1: Clock supply to TSG20 is stopped.

12.3.5.2 MSRTAPA — Module Standby Register – TAPA

Access: This register can be read/written in 8-bit units.

Address: FFC0 5004

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	MS_TAPA3	MS_TAPA2	—	—
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R	R

Table 12.6 MSRTAPA Register Contents

Bit Position	Bit Name	Function
7 to 4	—	Reserved The read value is always 0. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
3	MS_TAPA3	Setting this bit to 1 stops clock supply to TAPA3. 0: TAPA3 is operating. 1: Clock supply to TAPA3 is stopped.
2	MS_TAPA2	Setting this bit to 1 stops clock supply to TAPA2. 0: TAPA2 is operating. 1: Clock supply to TAPA2 is stopped.
1, 0	—	Reserved The read value is always 0. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.

12.3.5.3 MSROSTM — Module Standby Register – OSTM

Access: This register can be read/written in 8-bit units.

Address: FFC0 5008

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	MS_OSTM2	MS_OSTM1	MS_OSTM0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W

Table 12.7 MSROSTM Register Contents

Bit Position	Bit Name	Function
7 to 3	—	Reserved The read value is always 0. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
2	MS_OSTM2	Setting this bit to 1 stops clock supply to OSTM2. 0: OSTM2 is operating. 1: Clock supply to OSTM2 is stopped.
1	MS_OSTM1	Setting this bit to 1 stops clock supply to OSTM1. 0: OSTM1 is operating. 1: Clock supply to OSTM1 is stopped.
0	MS_OSTM0	Setting this bit to 1 stops clock supply to OSTM0. 0: OSTM0 is operating. 1: Clock supply to OSTM0 is stopped.

12.3.5.4 MSRWDTA — Module Standby Register – WDTA

Access: This register can be read/written in 8-bit units.

Address: FFC0 500C

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	MS_WDTA1	MS_WDTA0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 12.8 MSRWDTA Register Contents

Bit Position	Bit Name	Function
7 to 2	—	Reserved The read value is always 0. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
1	MS_WDTA1	Setting this bit to 1 stops clock supply to WDTA1. 0: WDTA1 is operating. 1: Clock supply to WDTA1 is stopped.
0	MS_WDTA0	Setting this bit to 1 stops clock supply to WDTA0. 0: WDTA0 is operating. 1: Clock supply to WDTA0 is stopped.

12.3.5.5 MSRPIC — Module Standby Register – PIC

Access: This register can be read/written in 8-bit units.

Address: FFC0 5010

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	MS_PIC2	MS_PIC1
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 12.9 MSRPIC Register Contents

Bit Position	Bit Name	Function
7 to 2	—	Reserved The read value is always 0. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
1	MS_PIC2	Setting this bit to 1 stops clock supply to PIC2. 0: PIC2 is operating. 1: Clock supply to PIC2 is stopped.
0	MS_PIC1	Setting this bit to 1 stops clock supply to PIC1. 0: PIC1 is operating. 1: Clock supply to PIC1 is stopped.

12.3.5.6 MSRRCAN — Module Standby Register – RS-CAN

Access: This register can be read/written in 8-bit units.

Address: FFC0 5014

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	MS_RCAN
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 12.10 MSRRCAN Register Contents

Bit Position	Bit Name	Function
7 to 1	—	Reserved The read value is always 0. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
0	MS_RCAN	Setting this bit to 1 stops clock supply to RS-CAN. 0: RS-CAN is operating. 1: Clock supply to RS-CAN is stopped.

12.3.5.7 MSRFRAY — Module Standby Register – FlexRay

Access: This register can be read/written in 8-bit units.

Address: FFC0 5018

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	MS_FRAY	—	—	—	MS_DSAD2_7
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R	R	R	R/W

Table 12.11 MSRFRAY Register Contents

Bit Position	Bit Name	Function
7 to 5	—	Reserved The read value is always 0. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
4	MS_FRAY	Setting this bit to 1 stops clock supply to FlexRay. 0: FlexRay is operating. 1: Clock supply to FlexRay is stopped.
3 to1	—	Reserved The read value is always 0. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
0	MS_DSAD2_7	Setting this bit to 1 stops clock supply to channels 2 to 7 of ΔΣAD. 0: ΔΣAD is operating. 1: Clock supply to channels 2 to 7 is stopped.

12.3.5.8 MSRRLIN — Module Standby Register – RLIN2

Access: This register can be read/written in 8-bit units.

Address: FFC0 501C

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	MS_RLIN0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 12.12 MSRRLIN Register Contents

Bit Position	Bit Name	Function
7 to 1	—	Reserved The read value is always 0. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
0	MS_RLIN0	Setting this bit to 1 stops clock supply to RLIN0. 0: RLIN0 is operating. 1: Clock supply to RLIN0 is stopped.

12.3.5.9 MSRSCI — Module Standby Register – SCI3

Access: This register can be read/written in 8-bit units.

Address: FFC0 5020

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	MS_SCI3	MS_SCI2	MS_SCI1	MS_SCI0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 12.13 MSRSCI Register Contents

Bit Position	Bit Name	Function
7 to 4	—	Reserved The read value is always 0. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
3	MS_SCI3	Setting this bit to 1 stops clock supply to SCI3. 0: SCI3 is operating. 1: Clock supply to SCI3 is stopped.
2	MS_SCI2	Setting this bit to 1 stops clock supply to SCI2. 0: SCI2 is operating. 1: Clock supply to SCI2 is stopped.
1	MS_SCI1	Setting this bit to 1 stops clock supply to SCI1. 0: SCI1 is operating. 1: Clock supply to SCI1 is stopped.
0	MS_SCI0	Setting this bit to 1 stops clock supply to SCI0. 0: SCI0 is operating. 1: Clock supply to SCI0 is stopped.

12.3.5.10 MSRCSIH — Module Standby Register – CSIH

Access: This register can be read/written in 8-bit units.

Address: FFC0 5024

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	MS_CSIH3	MS_CSIH2	MS_CSIH1	MS_CSIH0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 12.14 MSRCSIH Register Contents

Bit Position	Bit Name	Function
7 to 4	—	Reserved The read value is always 0. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
3	MS_CSIH3	Setting this bit to 1 stops clock supply to CSIH3. 0: CSIH3 is operating. 1: Clock supply to CSIH3 is stopped.
2	MS_CSIH2	Setting this bit to 1 stops clock supply to CSIH2. 0: CSIH2 is operating. 1: Clock supply to CSIH2 is stopped.
1	MS_CSIH1	Setting this bit to 1 stops clock supply to CSIH1. 0: CSIH1 is operating. 1: Clock supply to CSIH1 is stopped.
0	MS_CSIH0	Setting this bit to 1 stops clock supply to CSIH0. 0: CSIH0 is operating. 1: Clock supply to CSIH0 is stopped.

12.3.5.11 MSRSAD — Module Standby Register – SAR AD

Access: This register can be read/written in 8-bit units.

Address: FFC0 5028

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	MS_SARAD1	MS_SARAD0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 12.15 MSRSAD Register Contents

Bit Position	Bit Name	Function
7 to 2	—	Reserved The read value is always 0. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
1	MS_SARAD1	Setting this bit to 1 stops clock supply to SAR-AD1. 0: SAR-AD1 is operating. 1: Clock supply to SAR-AD1 is stopped.
0	MS_SARAD0	Setting this bit to 1 stops clock supply to SAR-AD0. 0: SAR-AD0 is operating. 1: Clock supply to SAR-AD0 is stopped.

12.3.5.12 MSRDAD — Module Standby Register – ΔΣAD

Access: This register can be read/written in 8-bit units.

Address: FFC0 502C

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	MS_DSAD
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 12.16 MSRDAD Register Contents

Bit Position	Bit Name	Function
7 to 1	—	Reserved The read value is always 0. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
0	MS_DSAD	Setting this bit to 1 stops clock supply to ΔΣAD. 0: ΔΣAD is operating. 1: Clock supply to ΔΣAD is stopped.

12.3.5.13 MSRATU — Module Standby Register – ATU-IV

Access: This register can be read/written in 8-bit units.

Address: FFC0 5030

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	MS_ATU
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 12.17 MSRATU Register Contents

Bit Position	Bit Name	Function
7 to 1	—	Reserved The read value is always 0. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
0	MS_ATU	Setting this bit to 1 stops clock supply to ATU. 0: ATU is operating. 1: Clock supply to ATU is stopped.

12.3.5.14 MSRAPA — Module Standby Register – APA

Access: This register can be read/written in 8-bit units.

Address: FFC0 5034

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	MS_APA
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 12.18 MSRAPA Register Contents

Bit Position	Bit Name	Function
7 to 1	—	Reserved The read value is always 0. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
0	MS_APA	Setting this bit to 1 stops clock supply to APA. 0: APA is operating. 1: Clock supply to APA is stopped.

12.3.5.15 MSRDFE — Module Standby Register – DFE

Access: This register can be read/written in 8-bit units.

Address: FFC0 5038

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	MS_DFE
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 12.19 MSRDFE Register Contents

Bit Position	Bit Name	Function
7 to 1	—	Reserved The read value is always 0. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
0	MS_DFE	Setting this bit to 1 stops clock supply to DFE. 0: DFE is operating. 1: Clock supply to DFE is stopped.

12.3.5.16 MSRRHSB — Module Standby Register – RHSB

Access: This register can be read/written in 8-bit units.

Address: FFC0 503C

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	MS_RHSB1	MS_RHSB0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 12.20 MSRRHSB Register Contents

Bit Position	Bit Name	Function
7 to 2	—	Reserved The read value is always 0. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
1	MS_RHSB1	Setting this bit to 1 stops clock supply to RHSB1. 0: RHSB1 is operating. 1: Clock supply to RHSB1 is stopped.
0	MS_RHSB0	Setting this bit to 1 stops clock supply to RHSB0. 0: RHSB0 is operating. 1: Clock supply to RHSB0 is stopped.

12.3.6 MSRPCMD — MSR Protection Command Register

The MSRPCMD register is used for protection against writing operation to the registers which may have a material effect on the system so that the application system is not incorrectly stopped due to program malfunction and the like.

The values of the module standby registers can be rewritten by a sequence specified in advance and unauthorized writing operation to them is eliminated.

Access: This register can be written in 8-bit units. The read value is undefined.

Address: FFC0 5070

Value after reset: Undefined

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—
Value after reset	—	—	—	—	—	—	—	—
R/W	W	W	W	W	W	W	W	W

Writing to the write-protected registers is only possible with the protection unlock sequence described below.

1. Write the fixed value ($A5_H$) to register MSRPCMD.
2. Write the new setting to the write-protected register. Write the value after a reset to the reserved bits.
3. Write the bitwise inverse of the setting value to the same register as in step 2. Write the inverse of the value after a reset to the reserved bits.
4. Write the new setting to the same register as in step 2. Write the value after a reset to the reserved bits.

A value can only be written to a write-protected register by following the procedure above.

Protection is not unlocked if the procedure is not followed correctly, the setting is not written to the write-protected register, and the MSRPS.MSRPRERR bit is set to 1.
(Although this is not a requirement, you can confirm if the values set in the targeted register was correctly written by checking that the setting of the MSRPS.MSRPRERR bit is 0 after step 4.)

In case of failure to follow the sequence, repeat the procedure from the beginning.

In case of writing to another register during steps 1 to 4 of the sequence, the protection function operates as follows. Operation is also as described when any interrupt is accepted during the sequence and the interrupt handling includes access to another register.

- When the writing is to another register in the same module*1, writing to the protected register fails and the MSRPS.MSRPRERR bit is set to 1.
- When writing is to a register in another module, writing to the protected register will successfully proceed to completion, if the remainder of the procedure is followed correctly.

Reading of another register while the sequence is in progress does not lead to failure.

Note 1. “Another register in the same module” indicates a register under the same module name as that containing the write-protected register. For the module names of the registers, see **Appendix, List of Registers**.

12.3.7 MSRPS — MSR Protection Status Register

The MSRPS register is the one to which the MSRPRERR flag indicating a protection error occurrence is allocated.

When an operation to write to the module standby register is not performed in a correct sequence including command register settings, such writing to the target register does not proceed. A protection error occurs and the status flag (MSRPRERR) is set to 1.

Access: This register can be read in 8-bit units.

Address: FFC0 5074

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	MSRPRERR
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 12.21 MSRPS Register Contents

Bit Position	Bit Name	Function
7 to 1	—	Reserved The read value is always 0.
0	MSRPRERR	Protection error detection 0: Protection error does not occur. 1: Protection error occurs.

Section 13 Clocked Serial Interface H (CSIH)

This section contains a generic description of the Clocked Serial Interface H (CSIH).

13.1 CSIH Features

Channels

This microcontroller has the following number of channels of the Clocked Serial Interface H.

Table 13.1 Channels of CSIH

Clocked Serial Interface H	RH850/E1M-S
Number of channels	4
Name	CSIH0, CSIH1, CSIH2, CSIH3

Index n

Throughout this section, the individual channels of a Clocked Serial Interface H is identified by the index “n” (n = 0 to 3), for example, CSIHnCTL0 for the CSIHn control register 0.

Index x

The Clocked Serial Interface H has up to 6 chip select signals. Throughout this section, the individual chip select signals are identified by the index “x”; thus a certain chip select signal is denoted as CSIHnTCSSx. The number of chip select signals for each channel of CSIH is given in the following table.

Table 13.2 Number of CSIH Chip Select Signals

CSIHn Channel	RH850/E1M-S Chip Select Index
CSIH0	CSIH0TCSSx (x = 0 to 5)
CSIH1	CSIH1TCSSx (x = 0 to 3)
CSIH2	CSIH2TCSSx (x = 0 to 3)
CSIH3	CSIH3TCSSx (x = 0 to 3)

Register addresses

All CSIHn register addresses are given as address offsets to the individual base address <CSIHn_base>.
The base address <CSIHn_base> of each CSIHn is listed in the following table.

Table 13.3 Register Base Addresses <CSIHn_base>

CSIHn Channel	<CSIHn_base> Address
CSIH0	FFD8 0000_H
CSIH1	FFD8 2000_H
CSIH2	FFD8 4000_H
CSIH3	FFD8 6000_H

Data consistency check

The following table lists the port pins on which CSIHnTSO pin functions are multiplexed and the availability of data consistency checking. See **Section 13.5.12, Error Detection** for details about data consistency checking.

Table 13.4 CSIHn Data Consistency Checking and Port Pins

CSIHn I/O Port Signal	Function	Alternative Function	Data Consistency Check
CSIH0			
CSIH0TSO	P0_1	ALT_OUT5	Available
CSIH1			
CSIH1TSO	P13_1	ALT_OUT6	Available
CSIH1TSO	P0_7	ALT_OUT6	Available
CSIH2			
CSIH2TSO	P4_8	ALT_OUT6	Available
CSIH2TSO	P0_12	ALT_OUT7	Available
CSIH3			
CSIH3TSO	P4_1	ALT_OUT7	Available
CSIH3TSO	P7_1	ALT_OUT6	Available

13.2 Notes on Pin Combination

CSIH uses multiple pins for one channel and the pins should be used in groups as given in the following table.

Group		CSIHn TSCK	CSIHn TSI	CSIHn TSO	CSIHn TCSS0	CSIHn TCSS1	CSIHn TCSS2	CSIHn TCSS3	CSIHn $\overline{\text{TSSI}}$	CSIHn TRY
CSIH1	A	P0_8	P0_6	P0_7	P0_9	P0_10	P0_11	P0_12	P0_13	P0_14
	B	P13_2	P13_0	P13_1	P13_3	P13_4	—	—	—	—
CSIH2	C	P4_9	P4_7	P4_8	P4_10	P4_11	P4_12	P4_13	P4_3	P4_4
	D	P0_13	P0_11	P0_12	P0_14	P1_0	P1_1	P1_2	P1_3	P1_4
CSIH3	E	P4_2	P4_0	P4_1	P4_3	P4_4	P4_5	P4_6	P4_7	P4_8
	F	P7_2	P7_0	P7_1	P7_3	P7_4	P7_5	P7_6	P7_7	P7_8

13.3 Function Overview

Overview of Functions

- Three-wire serial synchronous data transfer
- Master mode and slave mode selectable
- Multiple slaves configuration plus RCB (Recessive Configuration for Broadcasting) thanks to up to six configurable chip select output signals
- Built-in slave select function (slave select input signal ($\overline{\text{CSIHnTSSI}}$))
- Built-in baud rate generator
- Baud rate adjustable; in slave mode determined by input clock
- Maximum transmission speed:
  - in master mode: PCLK/8: 10 Mbps
  - in slave mode: PCLK/20: 4 Mbps
- Phase of clock and data selectable
- Data transfer with MSB or LSB first selectable
- Transfer data length selectable from 2 to 16 bits in 1-bit units
- EDL (Extended Data Length) function for transferring data with more than 16 bits
- Three selectable transfer modes:
  - transmit-only mode
  - receive-only mode
  - transmit/receive mode
- Built-in handshake function
- Error detection (data consistency check, parity, and overrun)
- Support of job concept
- JOB enable control bit for AUTOSAR is provided
- LBM (Loop Back Mode) function for self-test
- Enforced chip select idle setting

The block diagram shows the main components of the CSIH.

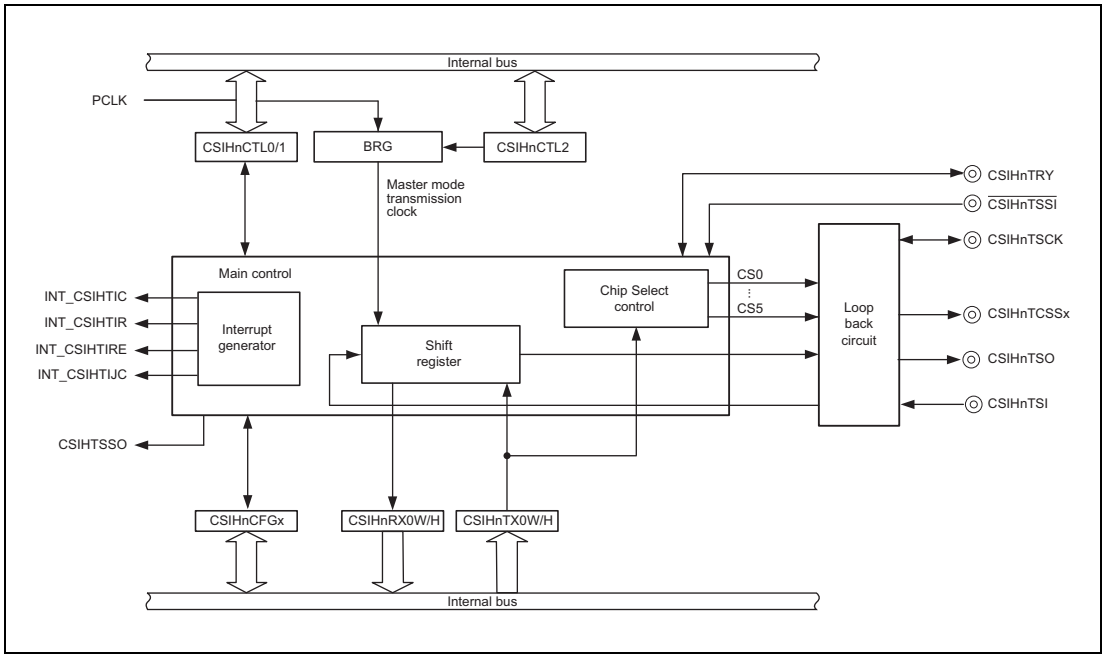


Figure 13.1 CSIH Block Diagram

In master mode, the transmission clock CSIHnTSCK is generated by the built-in baud rate generator (BRG). In slave mode, the transmission clock is provided from an external source.

The loop back circuit disconnects the CSIH completely from the ports and supports internal self-test.

CAUTION

This section describes the following modes:

- **The “operating mode” separates between master and slave mode. In this context, only a master can control and communicate with several slaves (for details see Section 13.5.1, Operating Modes (master/slave)).**
- **The “job mode” is related to the AUTOSAR job concept (for details see Section 13.5.3, Chip Selection (CS) Features, (3) Job concept).**
- **The “data transfer mode” specifies the kind of the communication – transmit-only, receive only, or transmit/receive (for details see Section 13.5.6, Data transfer modes).**

13.4 List of the Usage Notes

Table 13.5 Important Points for Register and Bit Operation (1/2)

Register name	Bit name	Contents
CSIHnCTL0	CSIHnPWR	If this bit is cleared during communication, ongoing communication is aborted.
CSIHnCTL0	CSIHnTXE CSIHnRXE	Do not modify any of these bits while CSIHnCTL0.CSIHnPWR = 0. (These bits and the CSIHnCTL0.CSIHnPWR bit can be modified at the same time.) Do not modify these bits while CSIHnSTR0.CSIHnTSF = 1 because ongoing communication is aborted and operation of this setting is not assured.
CSIHnCTL0	CSIHnJOBE	Do not modify this bit while CSIHnCTL0.CSIHnPWR = 0. This bit is only valid when CSIHnCTL1.CSIHnEDLE = 1. Setting this bit is prohibited in slave mode.
CSIHnCTL0	CSIHnMBS	Do not modify this bit while CSIHnCTL0.CSIHnPWR = 0. (This bit and the CSIHnCTL0.CSIHnPWR bit can be modified at the same time.) Modification of this bit is only allowed while CSIHnSTR0.CSIHnTSF = 0.
CSIHnCTL1	CSIHnCKR	Modification of the value of this bit is only permitted while CSIHnCTL0.CSIHnPWR = 0. If CS is not used, use this bit instead of CSIHnCFG0-5.CSIHnCKPx. CSIHnCFG0-5.CSIHnCKP must be set to 0. This bit must be used in slave mode.
CSIHnCTL1	CSIHnSLIT CSIHnCSL7-0 CSIHnDCS CSIHnCSRI CSIHnEDLE CSIHnHSE	Modification of the value of these bits is only permitted while CSIHnCTL0.CSIHnPWR = 0.
CSIHnCTL1	CSIHnJE CSIHnLBM	Modification of the value of these bits is only permitted while CSIHnCTL0.CSIHnPWR = 0. Setting these bits prohibited in slave mode.
CSIHnCTL1	CSIHnSSE	Modification of the value of this bit is only permitted while CSIHnCTL0.CSIHnPWR = 0. Setting this bit to 1 is prohibited in master mode.
CSIHnCTL1	CSIHnSIT	Modification of the value of this bit is only permitted while CSIHnCTL0.CSIHnPWR = 0. This bit is only valid in master mode. In slave mode, no delay is generated.
CSIHnCTL2	CSIHnPRS2-0	Modification of the value of these bits is only permitted while CSIHnCTL0.CSIHnPWR = 0. Max baud rate configuration is as follows. • Master mode: PCLK/8 • Slave mode: PCLK/20
CSIHnSTR0	CSIHnTSF	Writing is prohibited. Only reading is effective.
CSIHnSTR0	CSIHnDCE CSIHnPE CSIHnOVE	These bits are Initialized when CSIHnCTL0.CSIHnPWR = 0 →1 or CSIHnCTL0.CSIHnPWR = 1 → 0.
CSIHnCFGx	CSIHnBRSSx1-0 CSIHnRCBx CSIHnIDLx CSIHnIDx2-0 CSIHnHDx3-0 CSIHnINx3-0 CSIHnSPx3-0	Modification of the value of these bits is only permitted while CSIHnCTL0.CSIHnPWR = 0. Set these bits to 0 in slave mode.
CSIHnCFGx	CSIHnPSx1-0 CSIHnDLSx3-0 CSIHnDIRx CSIHnDAPx	Modification of the value of these bits is only permitted while CSIHnCTL0.CSIHnPWR = 0. In slave mode, the setting of the CSIHnCFG0 is used. Therefore, all these bits of CSIHnCFG 1 to 5 must be set to 0.

Table 13.5 Important Points for Register and Bit Operation (2/2)

Register name	Bit name	Contents
CSIHnCFGx	CSIHnCKPx	Modification of the value of this bit is only permitted while CSIHnCTL0.CSIHnPWR = 0. Since the CSIHnCTL1.CSIHnCKR bit is used in slave mode, set this bit to 0. If CS is not used, use CSIHnCTL1.CSIHnCKR instead of this bit and this bit must be set to 0.
CSIHnTX0W	CSIHnEOJ	This bit is only available while CSIHnCTL1.CSIHnEDLE = 1. When CSIHnCTL1.CSIHnJE = 0, the value is ignored even if the read value is 1. In slave mode, this bit must be set to “0”.
CSIHnTX0W	CSIHnEDL	This bit is only available while CSIHnCTL1.CSIHnEDLE = 1. When CSIHnCTL1.CSIHnEDLE = 0, the value is ignored even if the read value is 1.
CSIHnTX0W	CSIHnCS7-0	Setting these bits to FF_H is prohibited in master mode. These bits must be set to FE_H in slave mode.
CSIHnTX0W CSIHnTX0H		Write access to these bits is prohibited when CSIHnCTL0.CSIH0TXE = CSIHnCTL0.CSIH0RXE = 0.
CSIHnRX0W CSIHnRX0H		These bits are initialized when CSIHnCTL0.CSIHnPWR = 0 →1 or CSIHnCTL0.CSIHnPWR = 1 → 0.
CSIHnBRSi		Modification of the value of these bits is only permitted while CSIHnCTL0.CSIHnPWR = 0. The maximum baud rates are determined in combination with the setting of the CSIHnCTL2.PRS2-0 bits and are as follows. • Master mode: PCLK/8 • Slave mode: PCLK/20

13.5 Functional Description

The Clocked Serial Interface uses three signals for communication:

- Transmission clock CSIHnTSCK (output in master mode, input in slave mode)
- Data output signal CSIHnTSO
- Data input signal CSIHnTSI

Additional signals are available for external control and monitoring.

- $\overline{\text{CSIHnTSSI}}$: Slave select input signal
- CSIHnTRY: Handshake signal
- CSIHnTCSSx: Chip select signals

Data transmission is bit-wise and serial, and synchronous to the transmission clock.

The following table shows the most important registers for setting up the CSIH.

Table 13.6 Main Registers of CSIH

Register	Function
CSIHnCTL0	Enables/disables transmission clock, and permits/ prohibits data transmission and data reception. Defines end-of-job behavior and enables/disables buffering (bypass of the buffer).
CSIHnCTL1	Controls options like interrupt timing, extended data length, job feature, data consistency check, loop-back mode, handshake, etc.
CSIHnCTL2	Selects master/slave mode, and the baud rate of the internal generator (BRG) in master mode
CSIHnBRSi	Registers to configure the baud rate for each chip select signal
CSIHnCFGx	Registers to configure the communication protocol for each chip select signal

13.5.1 Operating Modes (master/slave)

For a particular CSIH module, the master or slave mode determines the source of the serial clock.

(1) Master mode

In master mode, the serial transmission clock is generated by the internal baud rate generator (BRG) and provided to the slave(s) by signal CSIHnTSCK.

Master mode is enabled by setting CSIHnCTL2.CSIHnPRS[2:0] to any value other than but 111_B. In master mode, the BRG frequency can be set by combining the CSIHnCTL2.CSIHnPRS[2:0] bits and the CSIHnBRSi.CSIHnBRSi[11:0] bits.

Chip select signals

In master mode, one or several chip select signals can be used. If several slaves are connected to the master, the chip select signals can be used to address one or several of the slaves. Only a selected slave is then enabled for communication.

The communication protocol and additional parameters are stored separately for each chip select signal. This makes it possible to adapt the data transfer individually to the requirements of each slave. For details, see **Section 13.5.3, Chip Selection (CS) Features**.

Clock default setting

The default level of CSIHnTSCK depends on the clock inverse bit CSIHnCTL1.CSIHnCKR of CSIHnTSCK: It is high level when CSIHnCTL1.CSIHnCKR = 0, and is low level when CSIHnCTL1.CSIHnCKR = 1.

The example below shows the communication in master mode for a data length of 8 bits, CSIHnCTL1.CSIHnCKR = 0, CSIHnCFGx.CSIHnCKPx = 0, CSIHnCFGx.CSIHnDAPx = 0, and MSB first:

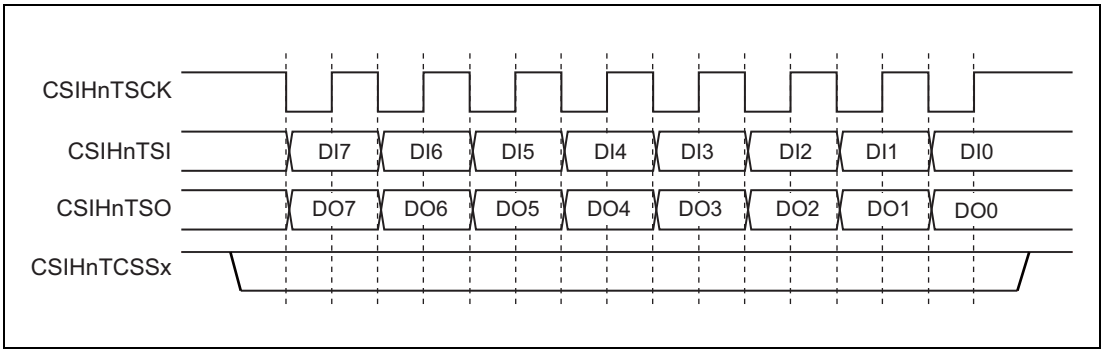


Figure 13.2 Transmission/Receive in Master Mode

(2) Slave mode

In slave mode, another device is the communication master and provides the transmission clock. Send/ receive operation normally starts as soon as a clock signal is detected.

Slave mode is selected by setting the CSIHnCTL2.CSIHnPRS[2:0] bits to 111_B.

In slave mode, the transmission protocol setting of the CSIHnCFG0 register is enabled (setting of the CSIHnCFG1 to CSIHnCFGx registers is disabled).

- CSIHnPS0[1:0]: parity usage
- CSIHnDLS0[3:0]: data length selection
- CSIHnDIR0: data direction
- CSIHnCKP0, CSIHnDAP0: clock and data phase

NOTE

When using slave mode, the baud rate generator (BRG) should be disabled by setting the CSIHnBRSi[11:0] bits in CSIHnBRSi to 000_H.

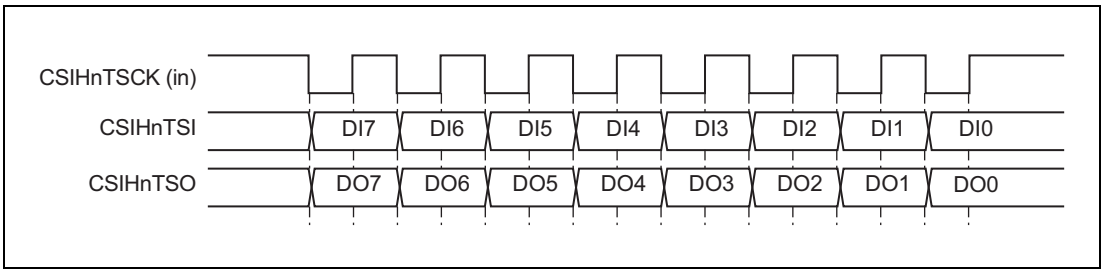


Figure 13.3 Transmission/Receive in Slave Mode

13.5.2 Master/Slave Connections

(1) One master and one slave

The following figure illustrates the connections between one master and one slave.

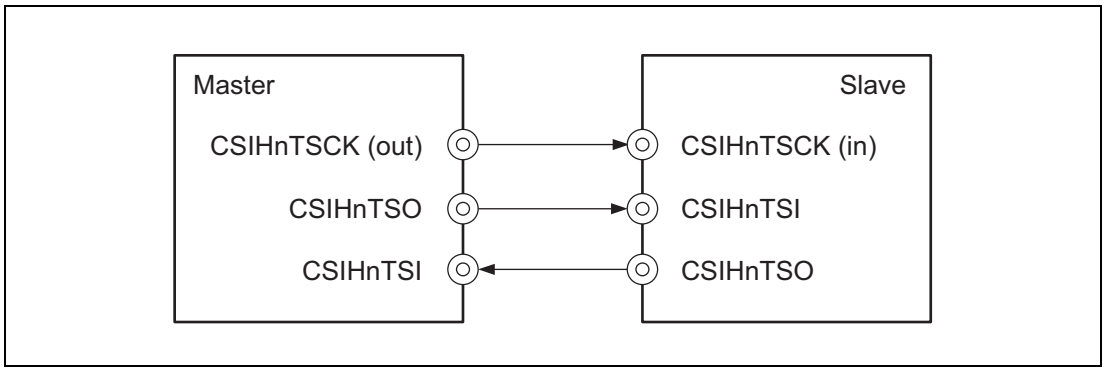


Figure 13.4 Direct Master/Slave Connection

(2) One master and multiple slaves

The following figure illustrates the connections between one master and multiple slaves. In this example, the master provides one chip select (CS) signal to each of the slaves. This signal is connected to the slave select input $\overline{\text{CSIHnTSSI}}$ of the slave.

The $\overline{\text{CSIHnTSSI}}$ signal can be enabled/disabled by the CSIHnCTL1.CSIHnSSE bit.

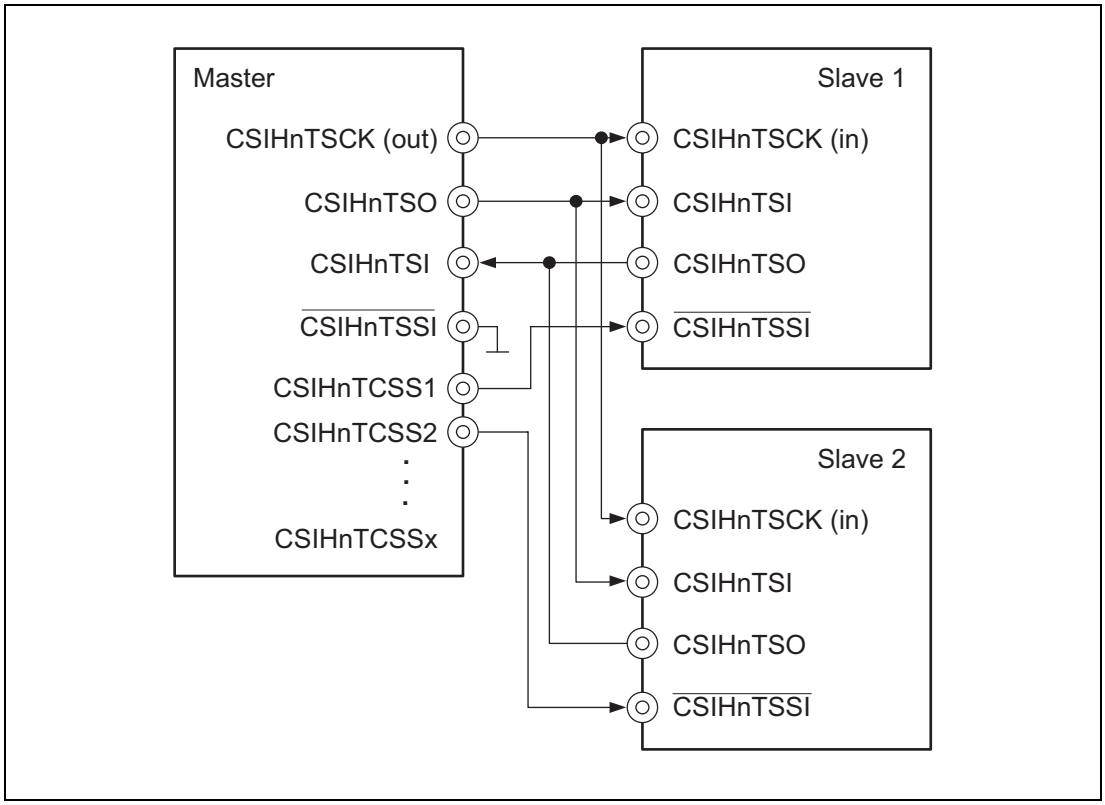


Figure 13.5 Master to Multiple Slaves Connection

By default, the chip select level is active low. That means, a slave is selected (enabled) as a CSIH slave when its $\overline{\text{CSIHnTSSI}}$ signal has low level. However, to adapt the CS to other devices, the output level of each chip select signal can also be programmed to be active high.

If a slave is not selected, it will neither receive nor transmit data. In addition, while transmit-only mode or transmit/receive mode is set (CSIHnCTL0.CSIHnTXE = 1), its output CSIHnTSO of a slave that is not selected is set to input mode in order to avoid interference with the output of another slave that was selected.

13.5.3 Chip Selection (CS) Features

The chip select signal, CSIHnTCSSx can be used by the master to select one or several slaves for communication.

(1) Configuration registers

The parameters for each chip select signal CSIHnTCSSx are defined in the corresponding configuration register CSIHnCFGx. The parameters include the communication protocol and additional CS parameters.

The communication protocol specifies:

- Data length: The number of bits to be sent or received.
- Transfer direction: MSB or LSB first.
- Parity usage: Odd, even, 0 parity or none.
- Clock phase and data phase.

Additional parameters for each chip select signal that is only available in master mode are:

- Prescaler selection of the baud rate generator separately for each chip select signal
- Chip select priority: Separates between “dominant” and “recessive” chip select signals. The priority applies if two or more chip selects signals with different configurations are simultaneously activated for message broadcasting. In this case, the configuration that is set as dominant is used.

The principle is also called “Recessive Configuration for Broadcasting” (RCB).

CAUTION

It is forbidden to specify several chip select signals as dominant with different configurations unless all dominant chip selects have the same configuration.

- Chip select timing:
  - Setup time T_{setup}: The time from setting the CS signal active to starting data output.
  - Inter-data time T_{inter}: The time between data and the next data while the same CS signal is active.
  - Hold time T_{hold}: Hold time of CS active level before changing the CS.
  - Idle time T_{idle}: Inactive time after terminating a CS signal or after every data transfer to the same CSx.

The CS timings of the setup time, the inter-data time, the hold time, and the idle time are illustrated in the figure below. When the CSIHnCFGx.CSIHnIDLx bit is set to 1, idle time is inserted every transfer regardless of the CS signal.

Figure 13.6 provides an example when the default CSIHnTCSS1 and CSIHnTCSS2 signals are active low (CSIHnCTL1.CSIHnCSL1 bit = 0, CSIHnCTL1.CSIHnCSL2 bit = 0). The active level can be specified individually for each CS.

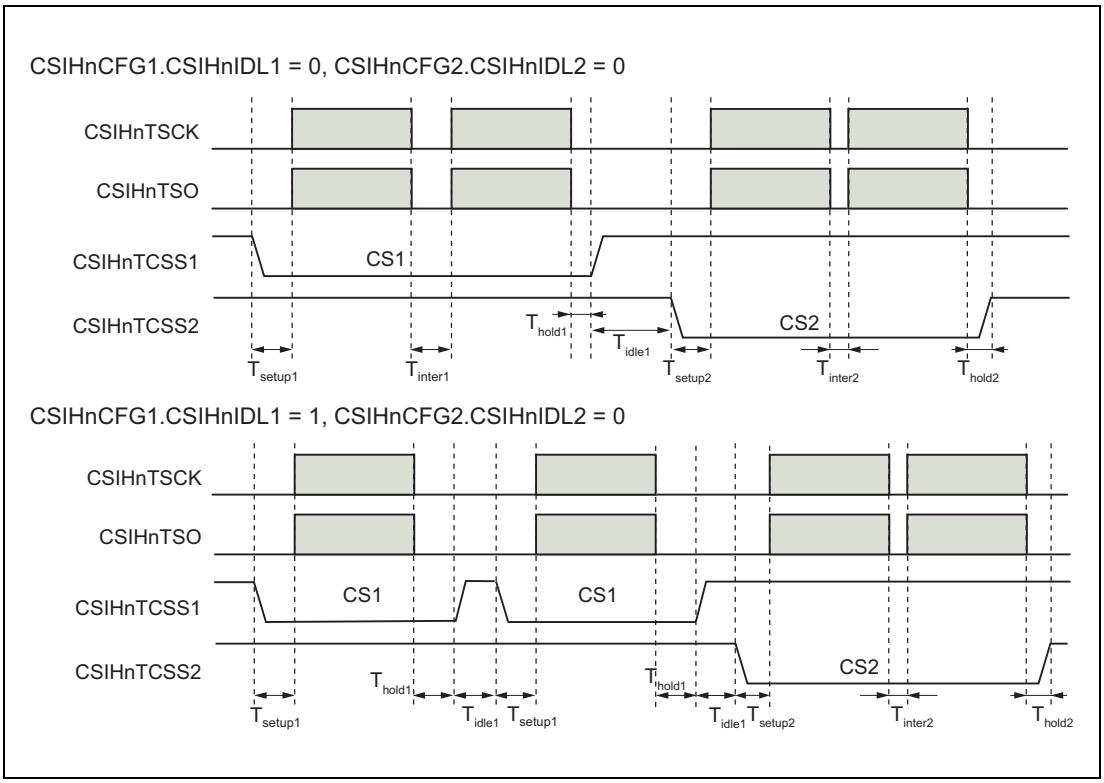


Figure 13.6 Chip Select Timings

Note that each CS signal can have a different value for setup, inter-data time, hold time, and idle time.

A particular chip select signal is activated by setting the appropriate bit in the transmission register CSIHnTX0W.CSIHnCS[7:0].

CSIHnRX0W.CSIHnCS[7:0] in the reception register indicates the chip select signal associated with the received data.

(2) CS example

The following figure shows an example of two consecutive data transmissions.

The first communication uses CS0 to communicate with one single slave. The second enables CS0 and CS1 to broadcast a message to two slaves. The priority of CS0 is set to “recessive: low priority”, the priority of CS1 to “dominant: high priority”.

Therefore, the second communication is conducted by using the CS1 setting specified in the dominant.

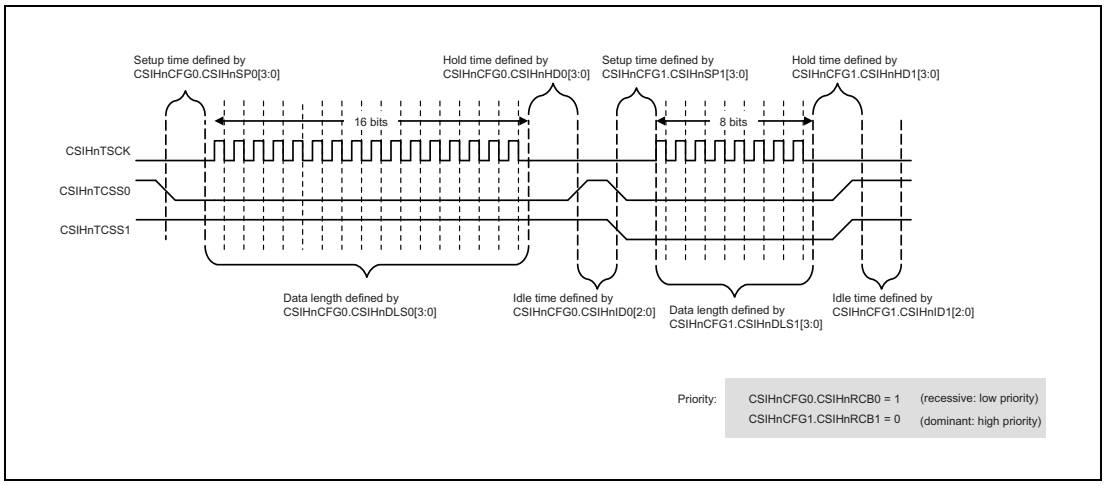


Figure 13.7 Chip Select and RCB Example

(3) Job concept

In terms of CSIH, a job consists of a number of data that are transferred.

Job mode enable

The job mode can only be enabled in master mode. The job mode is enabled and disabled by CSIHnCTL1.CSIHnJE, while the CSIH is disabled by CSIHnCTL0.CSIHnPWR = 0.

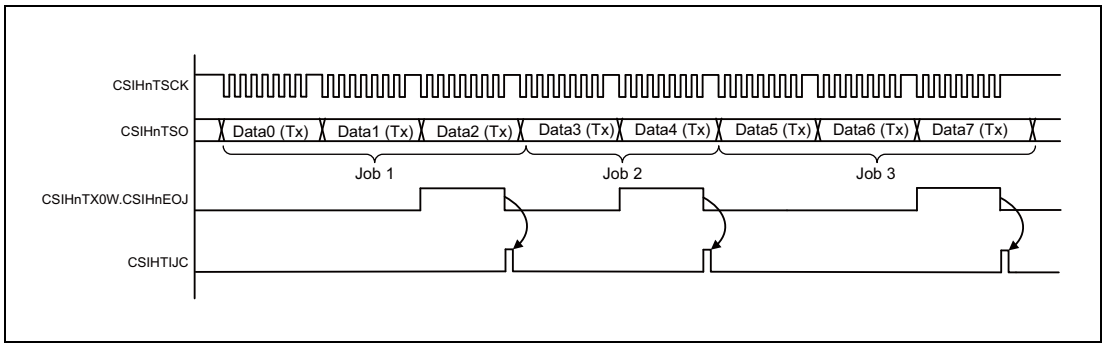


Figure 13.8 Job Examples

A job ends by transmitting data with the end-of-job bit set, i.e. data with CSIHnTX0W.CSIHnEOJ = 1.

A communication stop can be specified to occur after a job has finished. This is done by setting CSIHnCTL0.CSIHnJOBE. When CSIHnJOBE is set, the communication continues until data is sent, for which the CSIHnEOJ bit was set. After this data is sent, the communication is stopped and the job completion interrupt INT_CSIHTIJC is generated.

13.5.4 Chip Select Timing Details

(1) Changing the clock phase

The serial clock level specified by CSIHnCFGx.CSIHnCKPx may be changed when communication is disabled. The minimum value for idle time is 0.5 of a cycle of the transmission clock (CSIHnTSCK) when the clock phases are the same, and one cycle when the phases are different.

If the idle time is set to 0.5 transmission clock periods (in CSIHnCFGx.CSIHnIDx[2:0]) and two consecutive data are sent with different CSIHnCFGx.CSIHnCKPx configuration, the idle time is automatically extended to one period of CSIHnTSCK.

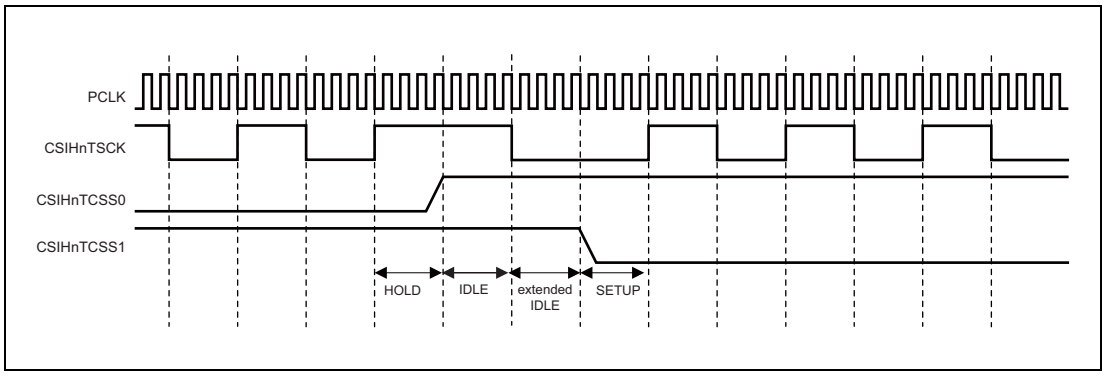


Figure 13.9 Clock Phase Timing
PCLK/8, Thold0 = Tsetup1 = 0 (0.5CSIHnTSCK), Tidle0 = 0 (0.5CSIHnTSCK), CSIHnCFG0.CSIHnCKP0 = 0 (CSIHnTCSS0) → CSIHnCFG1.CSIHnCKP1 = 1 (CSIHnTCSS1)

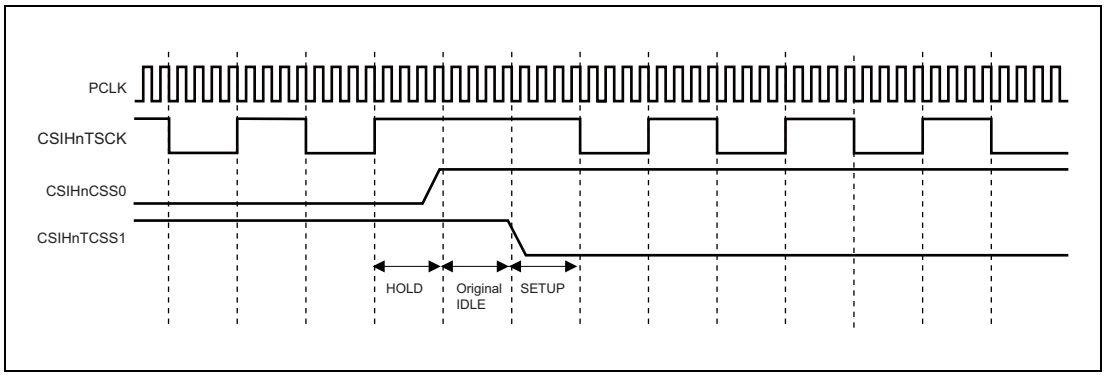


Figure 13.10 Clock Phase Timing
PCLK/8, Thold0 = Tsetup1 = 0 (0.5CSIHnTSCK), Tidle0 = 1 (1CSIHnTSCK), CSIHnCFG0.CSIHnCKP0 = 0 (CSIHnTCSS0) → CSIHnCFG1.CSIHnCKP1 = 1 (CSIHnTCSS1)

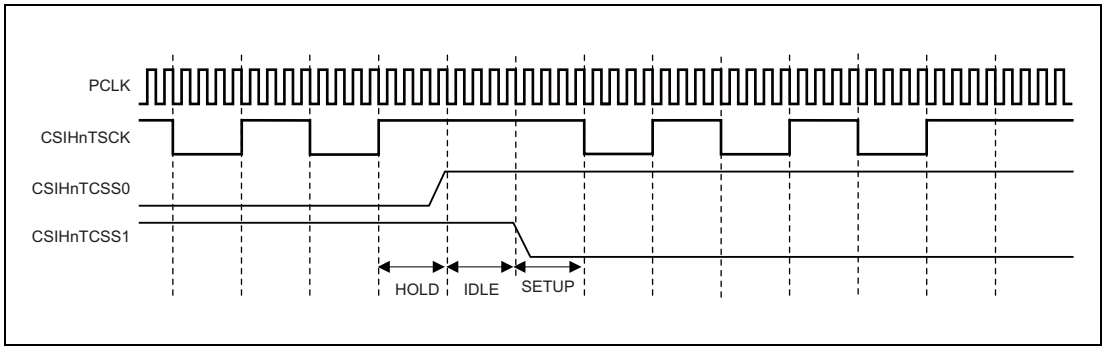


Figure 13.11 Clock Phase Timing
PCLK/8, Thold0 = Tsetup1 = 0 (0.5CSIHnTSCK), Tidle0 = 0 (0.5CSIHnTSCK), CSIHnCFG0.CSIHnCKP0 = 0 (CSIHnTCSS0)→ CSIHnCFG2.CSIHnCKP2 = 0 (CSIHnTCSS2)

(2) Changing the data phase

The CSIHnCFGx.CSIHnDAPx bit is used to define the phase of the data bit that is based on the clock.

The relationship between the CSIHnCFGx.CSIHnDAPx bit and hold/setup period is as followed.

The hold period is the time from the last edge of the serial clock (CSIHnTSCK) to the point where CSIHnTCSSx becomes inactive level.

The setup period is the time from the point where the CSIHnTCSSx becomes active level and to the point where transmit data (CSIHnTSO) is output.

Therefore, there is a 0.5 CSIHnTSCK gap until the edge of the serial clock (CSIHnTSCK) is output due to the setting of CSIHnCFGx.CSIHnDAPx.

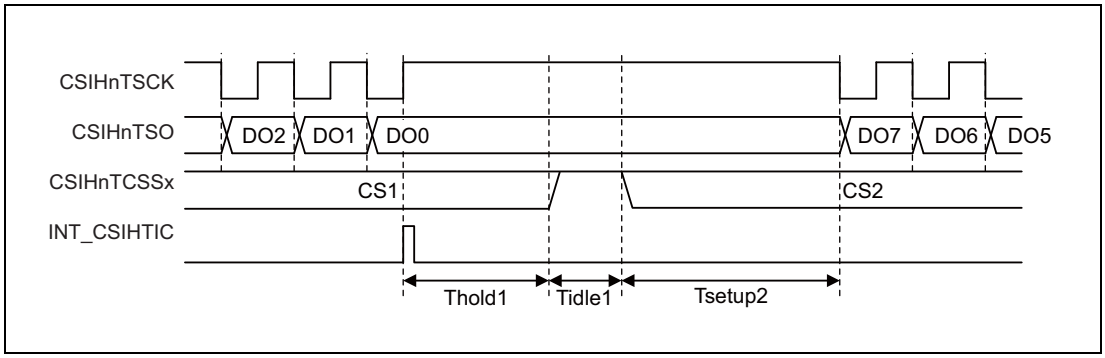


Figure 13.12 Data Phase Timing with CSIHnCFG1.CSIHnCKP1 = 0, CSIHnCFG1.CSIHnDAP1 = 0 and CSIHnCFG2.CSIHnCKP2 = 0, CSIHnCFG2.CSIHnDAP2 = 0

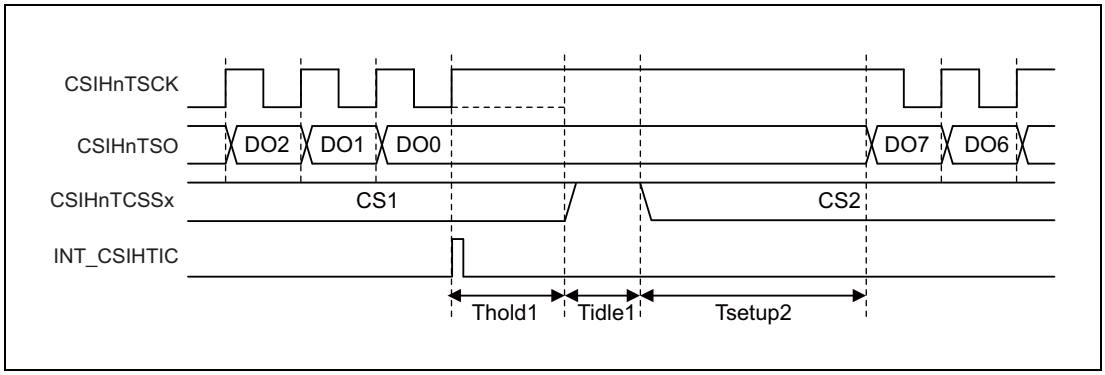


Figure 13.13 Data Phase Timing with CSIHnCFG1.CSIHnCKP1 = 1, CSIHnCFG1.CSIHnDAP1 = 0 and CSIHnCFG2.CSIHnCKP2 = 0, CSIHnCFG2.CSIHnDAP2 = 1

13.5.5 Transmission Clock Selection

In master mode, the transmission baud rate is selectable using the following bits:

- CSIHnCTL2.CSIHnPRS[2:0]
- CSIHnBRSi.CSIHnBRS[11:0]
- CSIHnCFGx.CSIHnBRSSx[1:0]

While the settings in the CSIHnCTL2.CSIHnPRS[2:0] and CSIHnBRSi.CSIHnBRSi[11:0] determine the baud rate of the transmission clock CSIHnTSCK, SIHnCFGx.CSIHnBRSSx[1:0] allows to select a baud rate setting for each chip select signal from the four settings of CSIHnBRS3 to CSIHnBRS0.

The following figure shows a block diagram of the baud rate generator.

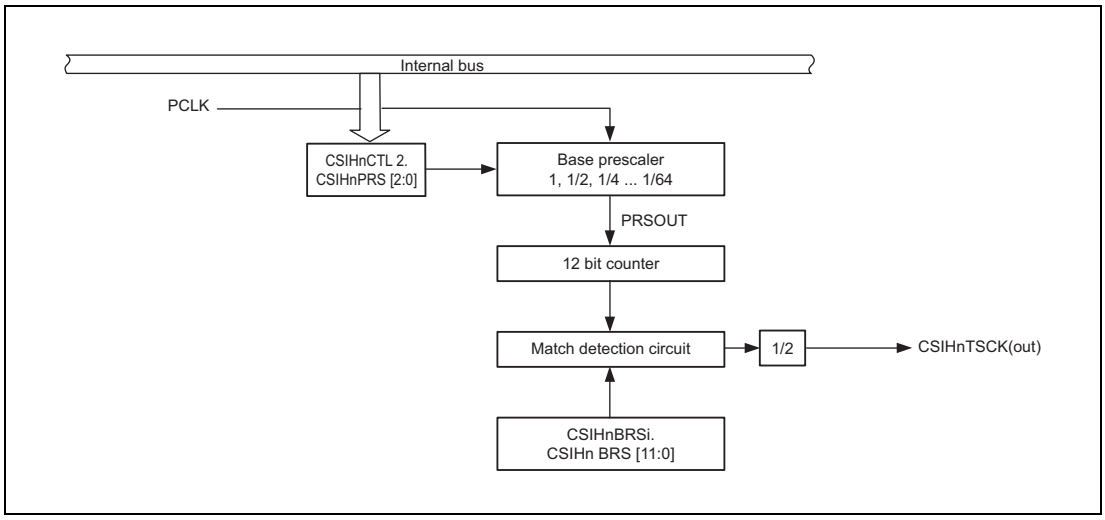


Figure 13.14 Baud Rate Generator Block Diagram

Setting CSIH0BRS[3:0].CSIHnBRS[11:0] to 000_H disables the baud rate generator, and thus all CSIHnTSCK are stopped.

Baud rate calculation

The baud rate is calculated as:

$$CSIHnTSCK = PCLK / (2^m \times k \times 2)$$

Where

m = CSIHnCTL2.CSIHnPRS[2:0] = 0 to 6

k = CSIH0BRSx (x = 0 to 3).CSIHBRS0 11 - 0 = 1 to 4095
(when CSIH0BRSS1 - 0 = 0)
CSIH0BRSx (x = 0 to 3).CSIHBRS1 11 - 0 = 1 to 4095
(when CSIH0BRSS1 - 0 = 1)
CSIH0BRSx (x = 0 to 3).CSIHBRS2 11 - 0 = 1 to 4095
(when CSIH0BRSS1 - 0 = 2)
CSIH0BRSx (x = 0 to 3).CSIHBRS3 11 - 0 = 1 to 4095
(when CSIH0BRSS1 - 0 = 3)

Baud rate limits

When setting the baud rate, please note:

- Maximum acceptable baud rate in master mode is PCLK/8.
- Maximum acceptable baud rate in slave mode is PCLK/20 (must be ensured by the external master).
- Minimum baud rate in both modes is PCLK/524160.

Example

If PCLK = 80 MHz, the maximum baud rate is as follows:

- 10 Mbps (PCLK/8) in master mode
- 4 Mbps (PCLK/20) in slave mode

The minimum baud rate is 152.625 bps (PCLK/524160).

13.5.6 Data transfer modes

(1) Transmit-only mode

Setting CSIHnCTL0.CSIHnTXE = 1 and CSIHnCTL0.CSIHnRXE = 0 puts the CSIH in transmit-only mode.

- Transmission starts when transmit data is written to the CSIHnTX0W or CSIHnTX0H register.

(2) Receive-only mode

Setting CSIHnCTL0.CSIHnTXE = 0 and CSIHnCTL0.CSIHnRXE = 1 puts the CSIH in receive-only mode.

- Receive starts when dummy data is written in the CSIHnTX0W or CSIHnTX0H register.

In slave mode, reception starts as soon as the CSIHnTSCK transmission clock from the master is received. It is not necessary to write data to the CSIHnTX0W or CSIHnTX0H register of the slave.

(3) Transmit/receive mode

Setting CSIHnCTL0.CSIHnTXE = 1 and CSIHnCTL0.CSIHnRXE = 1 puts the CSIH in transmit/receive mode.

- Communication starts when transmit data is written to the CSIHnTX0W or CSIHnTX0H register.

(4) Summary

The following table summarizes this section. It shows how data transfer is started in operating and transfer modes.

Table 13.7 Start of Data Transfer

Operating mode		Transfer mode	
		Transmit-only Transmit/receive	Receive-only
Direct access	Master	Writing to the CSIHnTX0W register or the CSIHnTX0H register	Writing to the CSIHnTX0W register or the CSIHnTX0H register
	Slave	Incoming clock from the master	Incoming clock from the master

13.5.7 Data Length Selection

(1) Data length between 2 and 16 bits

The length of a data packet is selectable for each chip select signal from 2 to 16 bits using CSIHnCFGx.CSIHnDLSx[3:0]. The examples below show the communication with MSB first (CSIHnCFGx.CSIHnDIRx = 0).

Data length = 16 bits (CSIHnCFGx.CSIHnDLSx[3:0] = 0000_B)

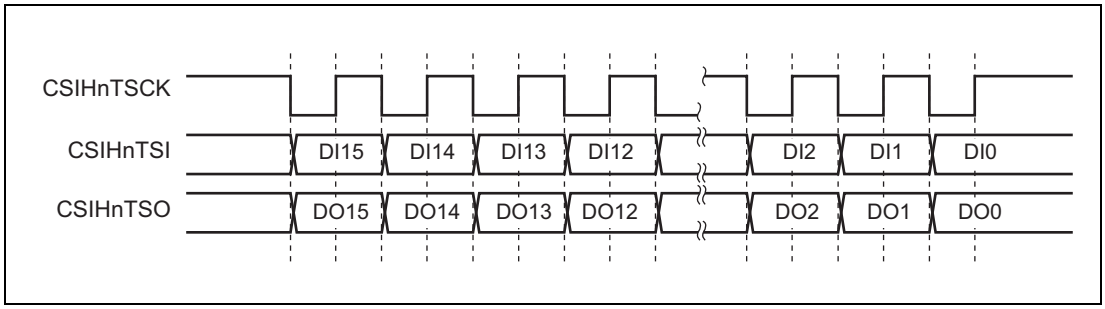


Figure 13.15 16 Bit Data Length, MSB First

Data length = 14 bits (CSIHnCFGx.CSIHnDLSx[3:0] = 1110_B):

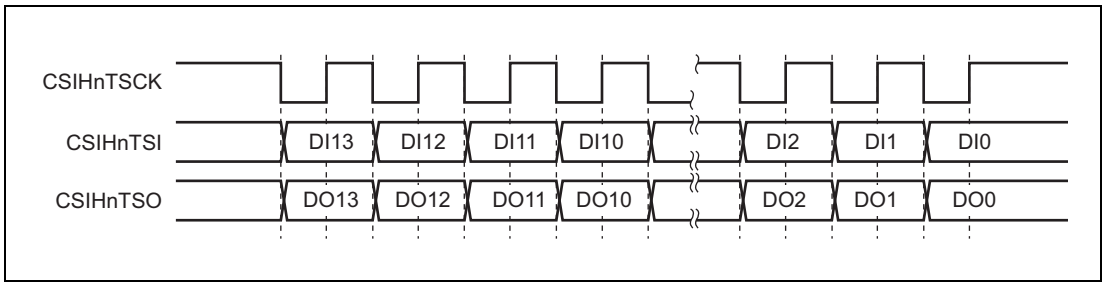


Figure 13.16 14 Bit Data Length, MSB First

(2) Data length greater than 16 bits

If the data to be sent/received exceeds 16 bits, the extended data length (EDL) feature can be used.

The EDL function is enabled by setting the CSIHnCTL1.CSIHnEDLE bit to 1.

The EDL function works as follows:

- The data has to be broken into 16-bit blocks plus remainder. For example, data of 42 bits would be broken into two 16-bit blocks plus 10 bits.
- The bit length of the reminder is set in CSIHnCFGx.CSIHnDLSx[3:0] as data length.
- For transmitting the 16-bit blocks, CSIHnTX0W.CSIHnEDL must be set to 1. In this case, the data written to CSIHnTX0W is sent as a 16-bit data length regardless of the CSIHnCFGx.CSIHnDLSx[3:0] bit setting.
- The transfer is complete after a block with the specified data length (the remainder of data specified with CSIHnTX0W.CSIHnEDL = 0) has been sent.

Example

Example for sending 40-bit data ($123456789A_H$) to CS0:

40 bits are split into two 16 bits and 8 bits.

- Initialize CSIHnCFG0.CSIHnDLS0[3:0] = 8.
- To send $123456789A_H$ with MSB first, write the following sequence to CSIHnTX0W:
  - $20FE\ 1234_H$ (CSIHnTX0W.CSIHnEDL = 1)
  - $20FE\ 5678_H$ (CSIHnTX0W.CSIHnEDL = 1)
  - $00FE\ 009A_H$ (CSIHnTX0W.CSIHnEDL = 0)

The following figure illustrates the timing.

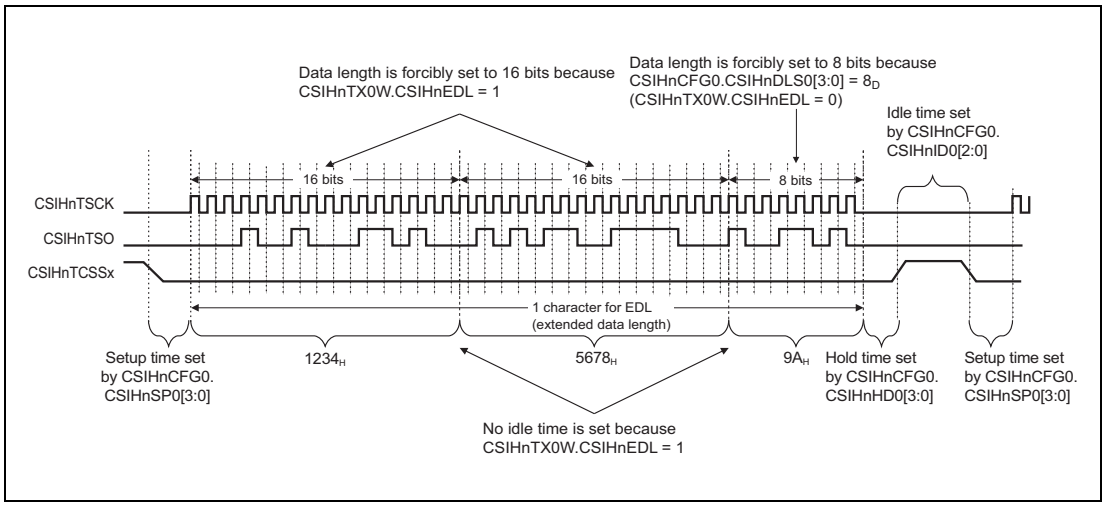


Figure 13.17 EDL Timing Diagram

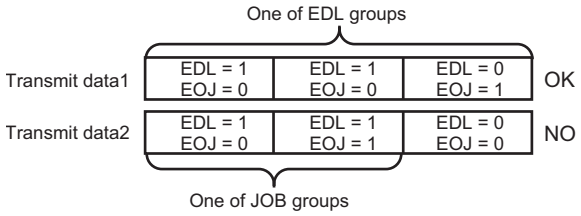
NOTES

1. If parity is enabled, the parity bit is added after the last bit.
2. When extended data length (EDL) mode is used for transmission, use the same chip select signal.
3. To consider the data direction, pay attention to the following example:

   Data to be sent: 123456_H

   MSB first:
   Set CSIHnCFGx.CSIHnDIRx = 0
   Write CSIHnTX0W = 2000 1234_H (EDL bit = 1)
   Write CSIHnTX0W = 0000 0056_H (EDL bit = 0)

   LSB first:
   Set CSIHnCFGx.CSIHnDIRx = 1
   Write CSIHnTX0W = 2000 3456_H (EDL bit = 1)
   Write CSIHnTX0W = 0000 0012_H (EDL bit = 0)
4. When EDL is enabled in job mode (CSIHnCTL1.CSIHnJE = 1 and CSIHnCTL1.CSIHnEDLE = 1), operation is not guaranteed if CSIHnTX0W.CSIHnEOJ is set to 1 while CSIHnTX0W.CSIHnEDL remains 1.



5. EDL mode cannot be used in receive-only mode in slave mode.(CSIHnCTL2.CSIHnPRS[2:0] = 111_B, CSIHnCTL0.CSIHnTXE = 0, CSIHnCTL0.CSIHnRXE = 1)

13.5.8 Serial Data Direction Selection

The serial data direction is selectable for each chip select signal using the CSIHnDIRx bit in the CSIHnCFGx register.

The examples below show communication for a data length of 8 bits (CSIHnCFGx.CSIHnDLSx[3:0] = 1000_B).

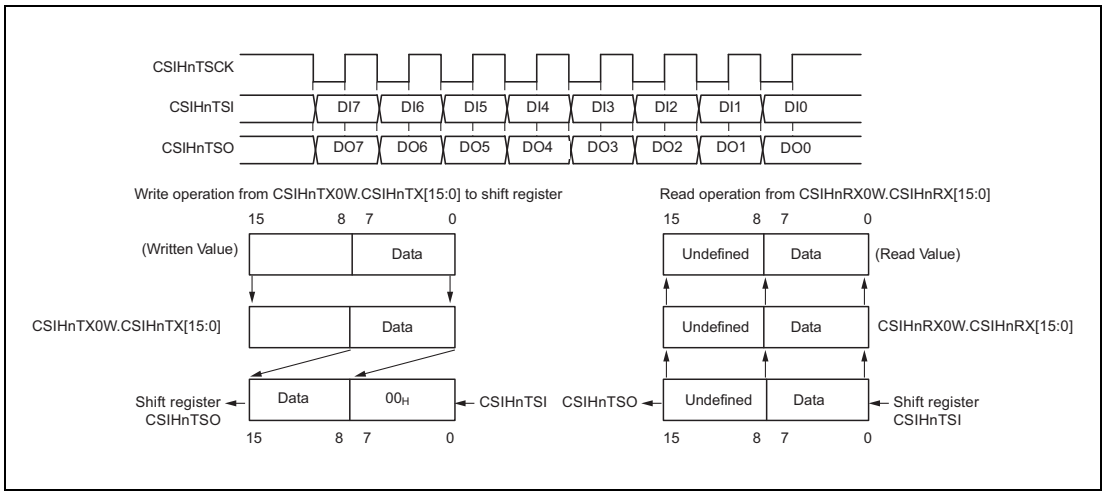


Figure 13.18 Serial Data Direction Select Function - MSB First (CSIHnDIRx = 0)

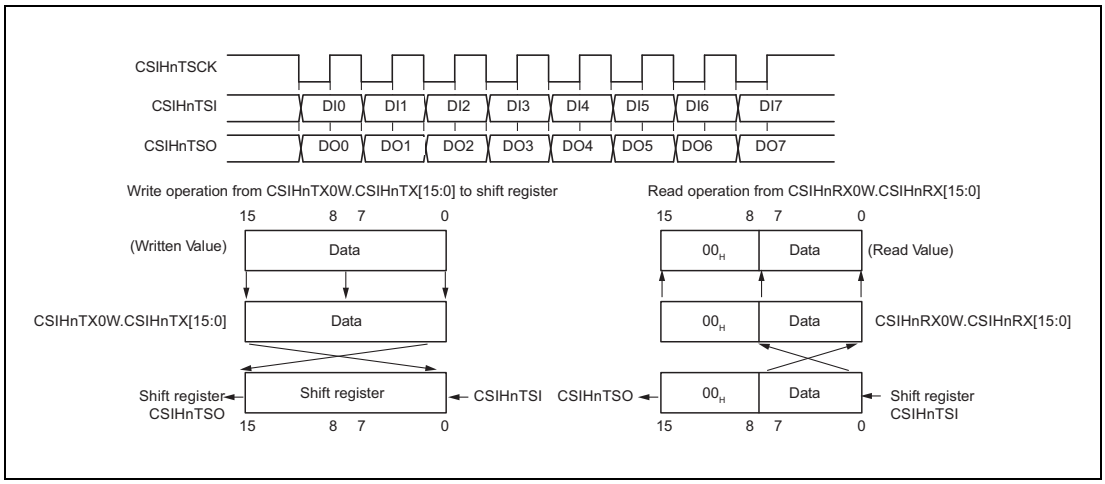
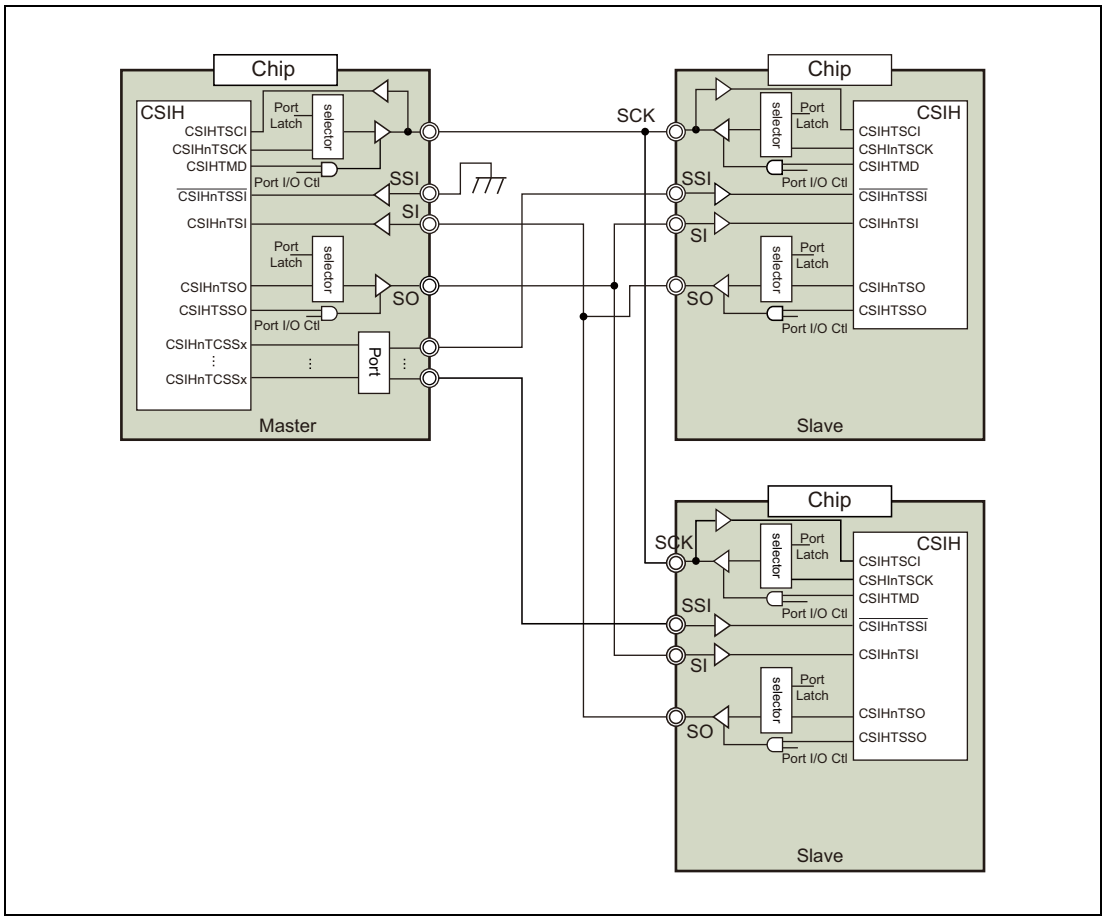


Figure 13.19 Serial Data Direction Select Function - LSB First (CSIHnDIRx = 1)

13.5.9 Communication in Slave Mode

(1) System outline



By the slave select function, the CSIH can realize one master and multiple slaves communication. The master chip outputs slave select signal (CSIHnTCSSx) to one of the slave chip. Each slave chip judges itself whether own CSIH is selected target slave or not. If selected, slave chip should control I/O function of SO pin to be an output pin. Then the slave chip can send transmission data to the master chip. If it is not selected as slave chip, SO pin should be an input function to avoid confliction of SO output from other selected slave chip. And also, not selected slave chip doesn't receive any data even if serial clock comes from master chip. By connecting master chip SI pin and multiple SO pins of all slave chips on a board, one master and multiple slaves communication can be realized. Putting level hold circuit to this serial data line is recommended.

(2) Communication timing

The following figure illustrates the communication signals and timings in slave mode.

In slave mode, the data transfer configuration is determined by the CSIHnCFG0 register.

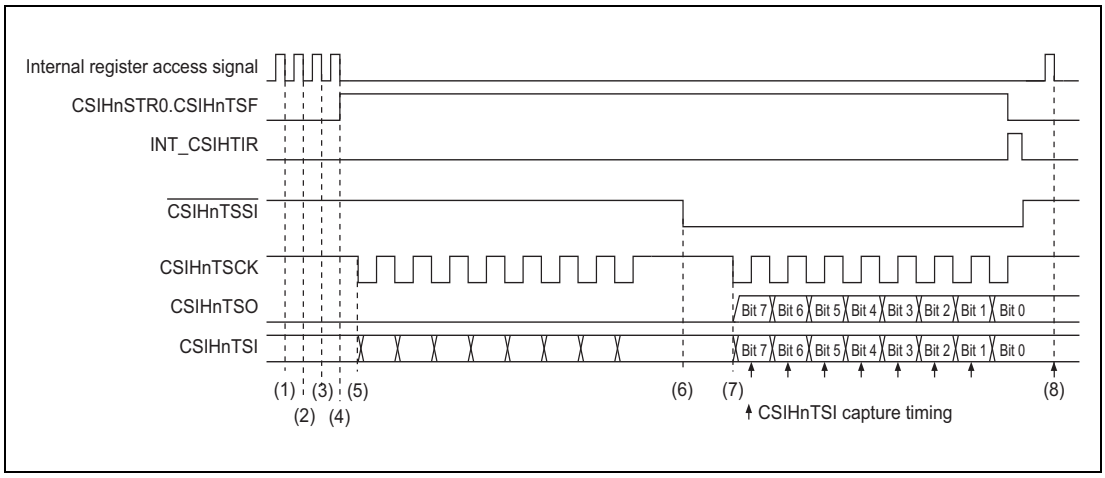


Figure 13.20 Transmit/Receive Communication Timing in Slave Mode

(1) CSIH is put into slave mode by setting CSIHnCTL2.CSIHnPRS[2:0] = 111_B. CSIHnCFG0.CSIHnCKP0 and CSIHnCFG0.CSIHnDAP0 are 0.

(2) The data length is 8 bits (CSIHnCFG0.CSIHnDLS0[3:0] = 1000_B). The data direction is MSB first (CSIHnCFG0.CSIHnDIR0 = 0).

(3) Transmit/receive mode is set (CSIHnCTL0.CSIHnCSIHnPWR = 1, CSIHnCTL0.CSIHnTXE = 1, CSIHnCTL0.CSIHnRXE = 1). Communication start is permitted.

(4) The transfer status flag CSIHnSTR0.CSIHnTSF is automatically set when transfer data is written to the CSIHnTX0W or CSIHnTX0H transmission register.

(5) As long as signal $\overline{\text{CSIHnTSSI}}$ is high, transmission/reception is not started, even if an external transmission clock CSIHnTSCK is applied. Input at CSIHnTSI is ignored.

(6) When $\overline{\text{CSIHnTSSI}}$ falls to low level, indicating that CSIHnTSO is enabled, transmission is enabled.

(7) On detection of the external clock signal CSIHnTSCK, the slave transmits data to CSIHnTSO and simultaneously captures data from CSIHnTSI.

(8) Interrupt INT_CSIHTIR indicates when the reception is complete. The CSIHnRX0W/H register can be read.

(3) Operation of CSIHTSSO

CSIHnPWR	CSIHnTXE	CSIHnRXE	CSIHnSSE	CSIHTSSO
0	—	—	—	H
1	—	—	0	H
	0		1	H
	1		1	Inverse of $\overline{\text{CSIHnTSSI}}$ level

The CSIHTSSO signal is a signal to control the I/O function of the CSIHnTSO pin in case of using the slave select function.

The CSIHnTSO pin is enabled when the CSIHTSSO signal is “High” (the CSIHnTSO pin is being driven).

The CSIHnTSO pin is disabled when the CSIHTSSO signal is “Low” (the CSIHnTSO pin is not being driven).

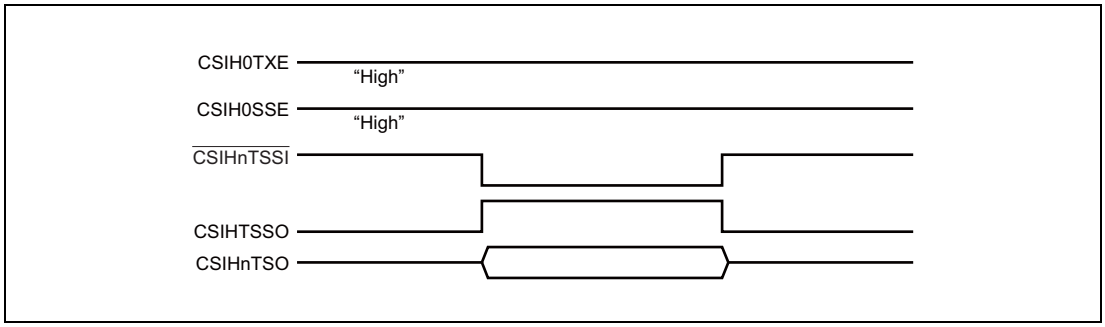


Figure 13.21 Operation of CSIHTSSO

CAUTION

If $\overline{\text{CSIHnTSSI}}$ pin is changed during communication (CSIH0STR0.CSIH0TSF = 1), current communication is not assured.

13.5.10 CSIH interrupt requests

CSIH can generate the following interrupt requests:

- INT_CSIHTIC (communication status interrupt)
- INT_CSIHTIR (reception status interrupt)
- INT_CSIHTIRE (communication error interrupt)
- INT_CSIHTIJC (job completion interrupt)

(1) Overview

The error interrupt INT_CSIHTIRE is generated when an error is detected. The generation of the other interrupts depends on the job mode, and – in case of the job completion interrupt INT_CSIHTIJC –also the operating mode.

The job completion interrupt INT_CSIHTIJC is only generated when job mode is enabled (CSIHnCTL1.CSIHnJE = 1). It is not available in slave mode.

The following table gives an overview.

Table 13.8 Interrupt Generation

Mode	Interrupt	Cause of interrupt	
		Job mode disabled CSIHnCTL1.CSIHnJE = 0	**Job mode enabled CSIHnCTL1.CSIHnJE = 1**
Direct access	INT_CSIHTIC (communication status interrupt)	Single data transfer	Single data transfer (except for job abortion*2)
	INT_CSIHTIR (reception status interrupt)	Data received and CSIHnCTL0.CSIHnRXE = 1	Data received *and* CSIHnCTL0.CSIHnRXE = 1
	INT_CSIHTIRE (communication error interrupt)	Error detected	Error detected
	INT_CSIHTIJC*1 (job completion interrupt)	Not applicable	Job abortion*2

Note 1. INT_CSIHTIJC is not available in slave mode.

Note 2. Job abortion condition: CSIHnTX0W.CSIHnEOJ = 1 and CSIHnCTL0.CSIHnJOBE = 1

(2) Interrupt delay

In master mode, all interrupts generated by the master can be delayed by one half period of the transmission clock, CSIHnTSCK. This is not possible in slave mode.

The delay is specified by setting CSIHnCTL1.CSIHnSIT = 1.

The following example illustrates the interrupt delay function, assuming a setting of CSIHnCTL1.CSIHnSIT = 1 (interrupt delay enabled), CSIHnCFGx.CSIHnCKPx = 0, CSIHnCFGx.CSIHnDAPx = 0 (normal clock and data phase), and CSIHnCFGx.CSIHnDLSx[3:0] = 1000_B (data length 8 bits).

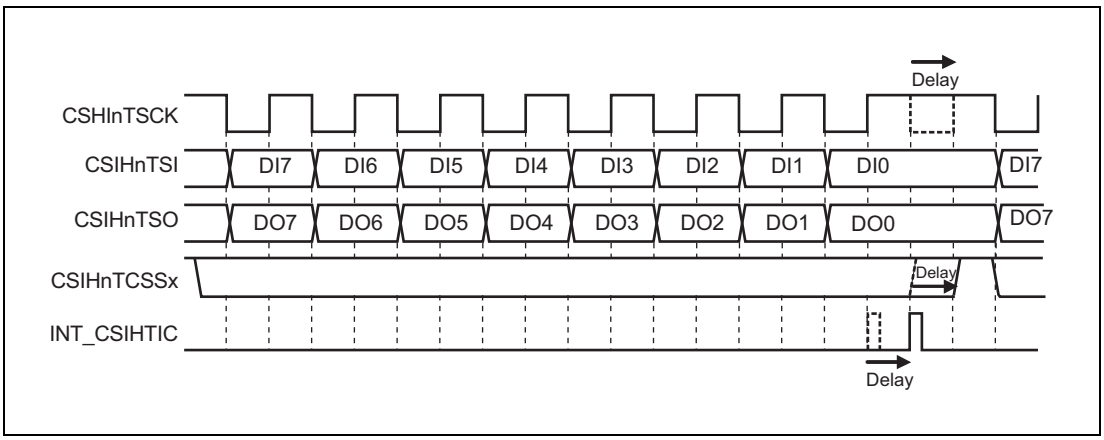


Figure 13.22 Interrupt Delay Function (CSIHnCTL1.CSIHnSIT = 1)

Setting CSIHnCTL1.CSIHnSIT = 1 adds half period delay to the transmission clock. This delays also the end of the present chip select signal (CSIHnTCSSx).

(3) INT_CSIHTIC (communication status interrupt)

Depending and the job mode, this interrupt is generated according to the conditions shown in the following table.

Table 13.9 INT_CSIHTIC Interrupt Generation

	Cause of interrupt	
Memory mode	**Job mode disabled CSIHnCTL1.CSIHnJE = 0**	**Job mode enabled CSIHnCTL1.CSIHnJE = 1**
Direct access	Generated after every data transfer.	Generated after every data transfer, except when the communication was aborted.

INT_CSIHTIC in direct access mode

The examples below show the INT_CSIHTIC behavior in direct access mode.

The examples assume:

- Master mode
- Direct access mode
- No interrupt delay (CSIHnCTL1.CSIHnSIT = 0)
- Normal clock and data phase (CSIHnCFGx.CSIHnCKPx = 0, CSIHnCFGx.CSIHnDAPx = 0)
- Data length 8 bits (CSIHnCFGx.CSIHnDLSx[3:0] = 1000_B)
- Normal INT_CSIHTIC interrupt timing (CSIHnCTL1.CSIHnSLIT = 0)

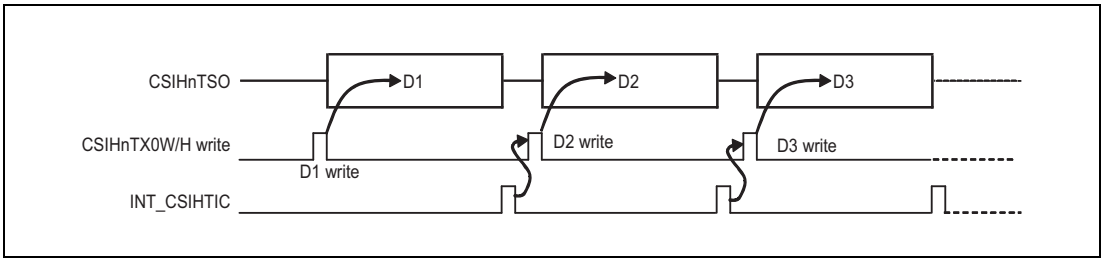


Figure 13.23 Generation of INT_CSIHTIC after Transfer (CSIHnCTL1.CSIHnSLIT = 0)

If job mode is enabled (CSIHnCTL1.CSIHnJE = 1) and a job ends because data is sent with CSIHnTXOW.CSIHnEOJ= 1 and communication stop is requested (CSIHnCTL0.CSIHnJOBE = 1), then INT_CSIHTIC is replaced by the job completion interrupt INT_CSIHTIC.

INT_CSIHTIC can also be set up to occur as soon as the CSIHnTX0W/H register is free for the next data. This is specified by setting CSIHnCTL1.CSIHnSLIT = 1.

The effect is illustrated in the figure below.

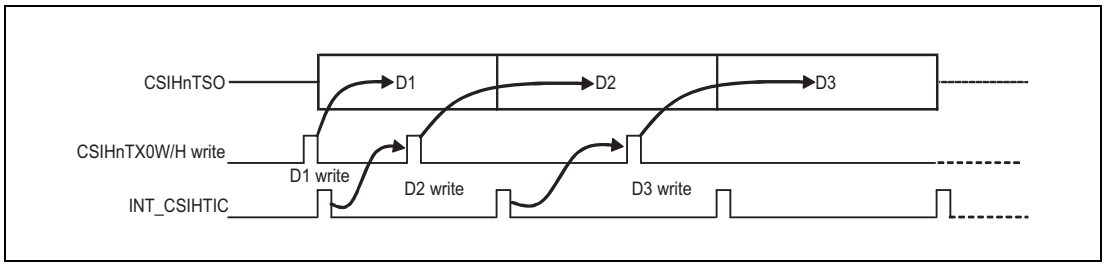


Figure 13.24 Immediate Generation of INT_CSIHTIC (CSIHnCTL1.CSIHnSLIT = 1)

Thus, the new data can be written in advance.

CAUTION

This mode allows faster data transfer but is only available in direct access mode.

INT_CSIHTIC in job mode

The example below shows the INT_CSIHTIC behavior in job mode.

The example assumes:

- Master mode
- Job mode enabled (CSIHnCTL1.CSIHnJE = 1)
- No interrupt delay (CSIHnCTL1.CSIHnSIT = 0)
- Normal clock and data phase (CSIHnCFGx.CSIHnCKPx = 0, CSIHnCFGx.CSIHnDAPx = 0)
- Data length 8 bits (CSIHnCFGx.CSIHnDLSx[3:0] = 1000_B)
- Normal INT_CSIHTIC interrupt timing (CSIHnCTL1.CSIHnSLIT = 0)

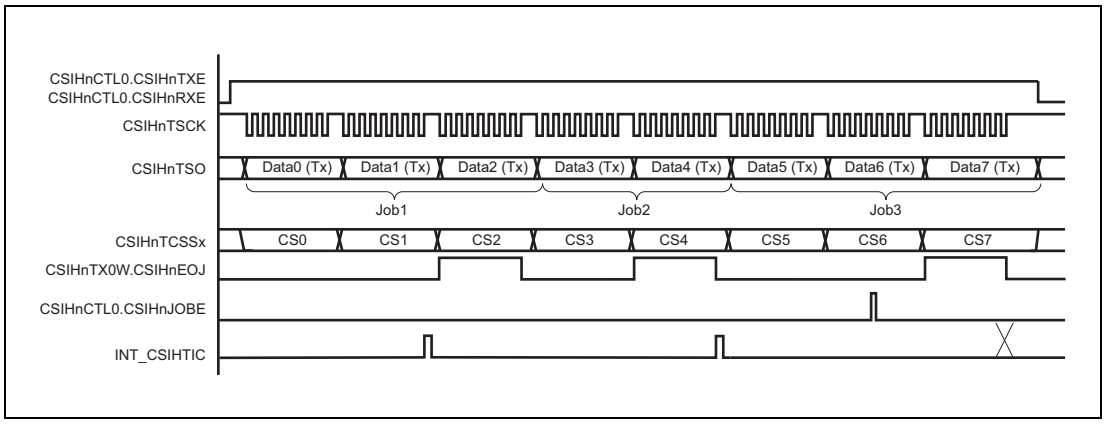


Figure 13.25 Generation of INT_CSIHTIC in Job Mode

The rules for generating INT_CSIHTIC in job mode are shown in the following table.

Table 13.10 Generation of INT_CSIHTIC in Job Mode

CSIHnTX0W. CSIHnEOJ	INT_CSIHTIC
0	Not generated
0	Generated
1	Not generated
1	CSIHnCTL0.CSIHnJOBE = 0: Generated
	CSIHnCTL0.CSIHnJOBE = 1: Not generated, replaced by interrupt INT_CSIHTIJC

(4) INT_CSIHTIR (reception status interrupt)

This interrupt is generated according to the conditions below.

Table 13.11 INT_CSIHTIR Interrupt Generation

Mode	Cause of interrupt	
	Job mode disabled CSIHnCTL1.CSIHnJE = 0	Job mode enabled CSIHnCTL1.CSIHnJE = 1
Direct access	Generated after every data transfer.	

INT_CSIHTIR in direct access mode

The example below shows the INT_CSIHTIR behavior in direct access mode.

The example below assumes:

- Master mode
- Direct access mode
- No interrupt delay (CSIHnCTL1.CSIHnSIT = 0)
- Normal clock and data phase (CSIHnCFGx.CSIHnCKPx = 0, CSIHnCFGx.CSIHnDAPx = 0)
- Data length 8 bits (CSIHnCFGx.CSIHnDLSx[3:0] = 1000_B)

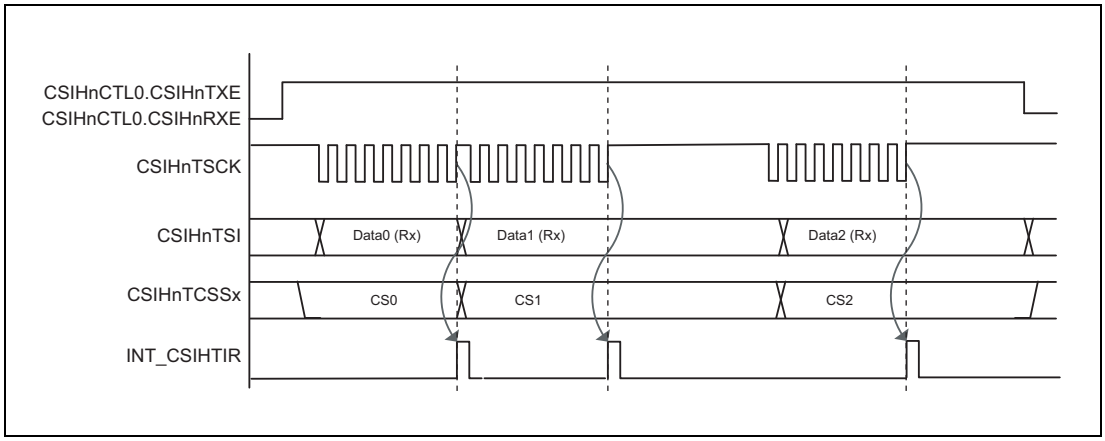


Figure 13.26 Generation of INT_CSIHTIR in Direct Access Memory Mode

(5) INT_CSIHTIRE (communication error interrupt)

This interrupt is generated whenever an error is detected.

For details about generation interruption timing, see **Section 13.5.12, Error Detection**.

Table 13.12 Data Error Types

Error type	Communication status after error interrupt	Comment
Parity error	Interrupt is generated and communication continues	—
Data consistency error	Interrupt is generated and communication continues	—
Overrun error	Interrupt is generated and communication continues	This error interrupt is only generated when CSIHnCTL1.CSIHnHSE = 0 (without handshake) in slave mode.

The type of error that caused the generation of INT_CSIHTIRE is flagged in register CSIHnSTR0.

Additionally a parity error flag and a data consistency error flag are attached to the received data in CSIHnRX0W.

For details about the various error types, see **Section 13.5.12, Error Detection**.

(6) INT_CSIHTIJC (job completion interrupt)

This interrupt supports the handling of jobs. See **Section 13.5.3 (3), Job concept**. This interrupt is only available in master mode.

Job mode is enabled by setting CSIHnCTL1.CSIHnJE = 1. When CSIHnCTL1.CSIHnJE = 0, INT_CSIHTIJC is not generated.

This interrupt is generated according to the condition shown in the following table.

Table 13.13 INT_CSIHTIJC Interrupt Generation

	Interrupt Sources	
Mode	Job mode disabled CSIHnCTL1.CSIHnJE = 0	Job mode enabled CSIHnCTL1.CSIHnJE = 1
Direct access	Not applicable	Indicates that the communication has stopped at the end of a job after a job abortion*1 was triggered

Note 1. Job abortion condition: CSIHnTX0W.CSIHnEOJ = 1 and CSIHnCTL0.CSIHnJOBE = 1

13.5.11 Handshake function

CSIH features a handshake function to synchronize the master and the slave devices. This function can be enabled/disabled by the CSIHnCTL1.CSIHnHSE bit. For handshake, the CSIHnTRY signal is used.

The busy timing depends on the setting of the data phase selection bit CSIHnCFGx.CSIHnDAPx.

(1) Slave mode

When CSIHnCTL1.CSIHnHSE = 1 and the slave is busy, the CSIHnTRY signal outputs low level (0). This can happen in the following case:

Case 1: When transmission is not ready (no next transmission data)

Case 2: When reception register is full:

Because the slave is in receive-only or transmit/receive mode but previously received data is still in the CSIHnRX0W/H register, new data cannot be copied from the shift register to CSIHnRX0W/H (CSIHnRX0W/H full condition).

The data length of 8 bits is assumed in the following example.

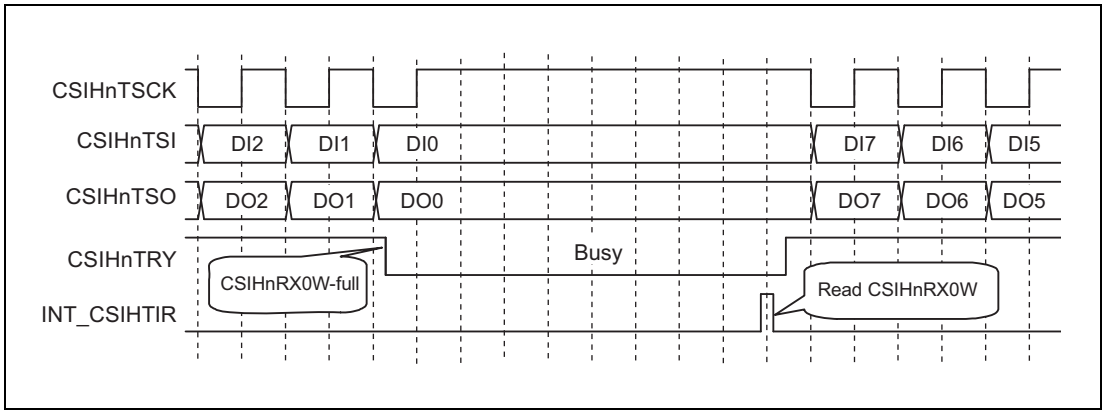


Figure 13.27 Busy Signal from the Slave (Direct Access Mode; CSIHnCFGx.CSIHnDAPx = 0)

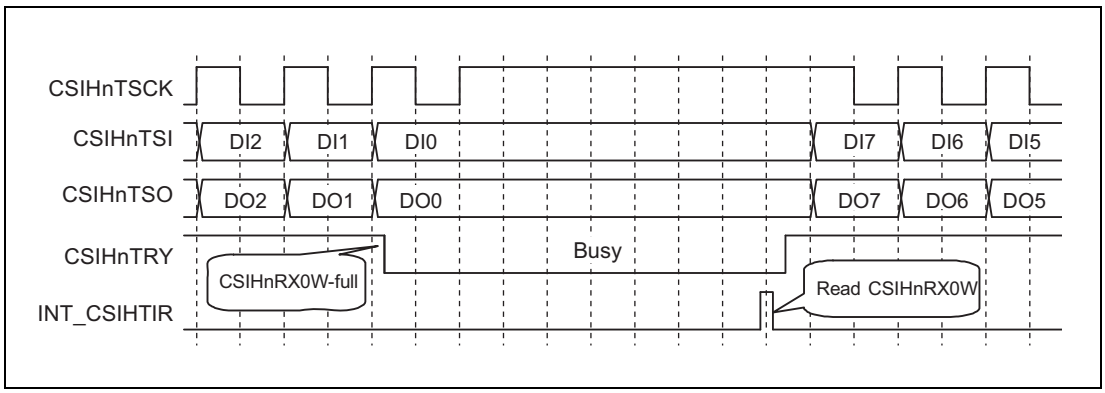


Figure 13.28 Busy Signal from the Slave (CSIHnCFGx.CSIHnDAPx = 1)

(2) Master mode

When the master detects CSIHnTRY being 0 while CSIHnCTL1.CSIHnHSE is 1, the subsequent transfer is put on hold and the master enters the waiting state. It suspends the CSIHnTSCK clock.

The CSIHnTRY level is checked at each half clock cycle of CSIHnTSCK.

The chip select signal is output during the waiting state.

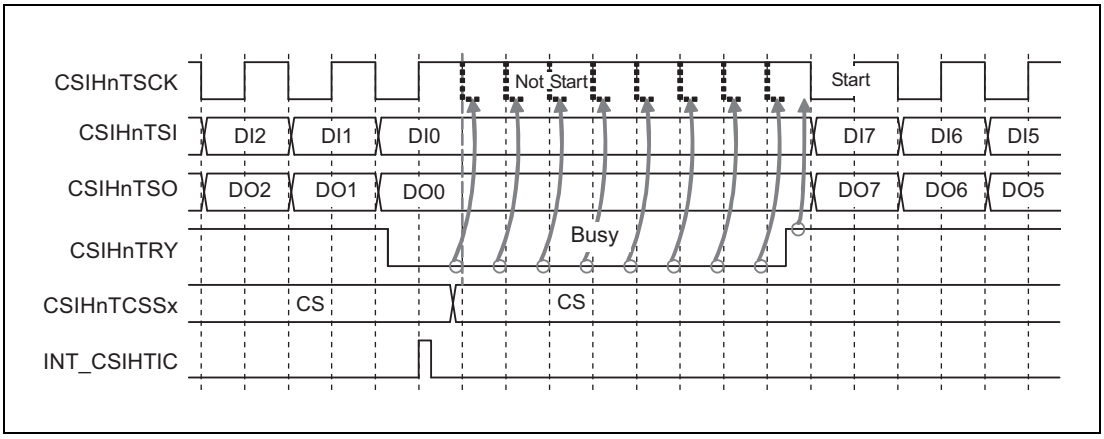


Figure 13.29 Master's Reaction on CSIHnTRY (CSIHnCFGx.CSIHnDAPx = 0)

CSIHnTRYI must be pulled down by the slave before the next transfer starts. When the slave changes the CSIHnTRYI signal to low level during data transfer, the serial clock from the master stops at the completion of data transfer.

The master resumes the communication as soon as CSIHnTRY becomes high level (the slave is "ready").

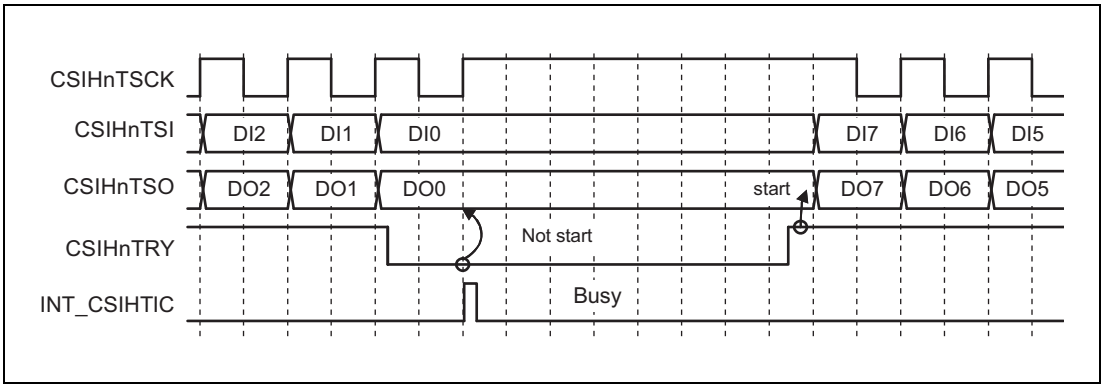


Figure 13.30 Master's Reaction on CSIHnTRY (CSIHnCFGx.CSIHnDAPx = 1)

CAUTIONS

1. **If multiple slaves are connected, the master must only detect the CSIHnTRY signal of the slave it has selected for communication.**
2. **Even when the slave changes the CSIHnTRY signal to low level during a transmission, the transmitting operation is continued until its successful completion.**

13.5.12 Error Detection

CSIH can detect three error types:

- Data consistency error (transmission data)
- Parity error (received data)
- Overrun error (received data)

Check for parity and data consistency can be enabled/disabled individually.

If one of these errors is detected, the interrupt request, INT_CSIHTIRE is generated and the corresponding flags are set.

(1) Data consistency check

The purpose of the data consistency check is to ensure that the data physically sent as output signal is identical to the original data that was copied to the shift register.

The data consistency check can be enabled/disabled by the CSIHnCTL1.CSIHnDCS bit. It is not active if data transmission is disabled (CSIHnCTL0.CSIHnTXE = 0).

When the data consistency check is active, the data transferred from CSIHnTX0W or CSIHnTX0H to the shift register is copied to a separate register. In addition, the physical levels at CSIHnTSO are read back via the CSIHTDCS signal into an own shift register.

After completion of the transmission, the sent data is compared with the original transmission data.

Mismatch is considered as a data consistency error:

- Interrupt INT_CSIHTIRE is generated.
- The CSIHnSTR0.CSIHnDCE bit is set.

Additionally, CSIHnRX0W.CSIHnTDCE of data that contains the error is set.

The data consistency check function is illustrated in the following block diagram.

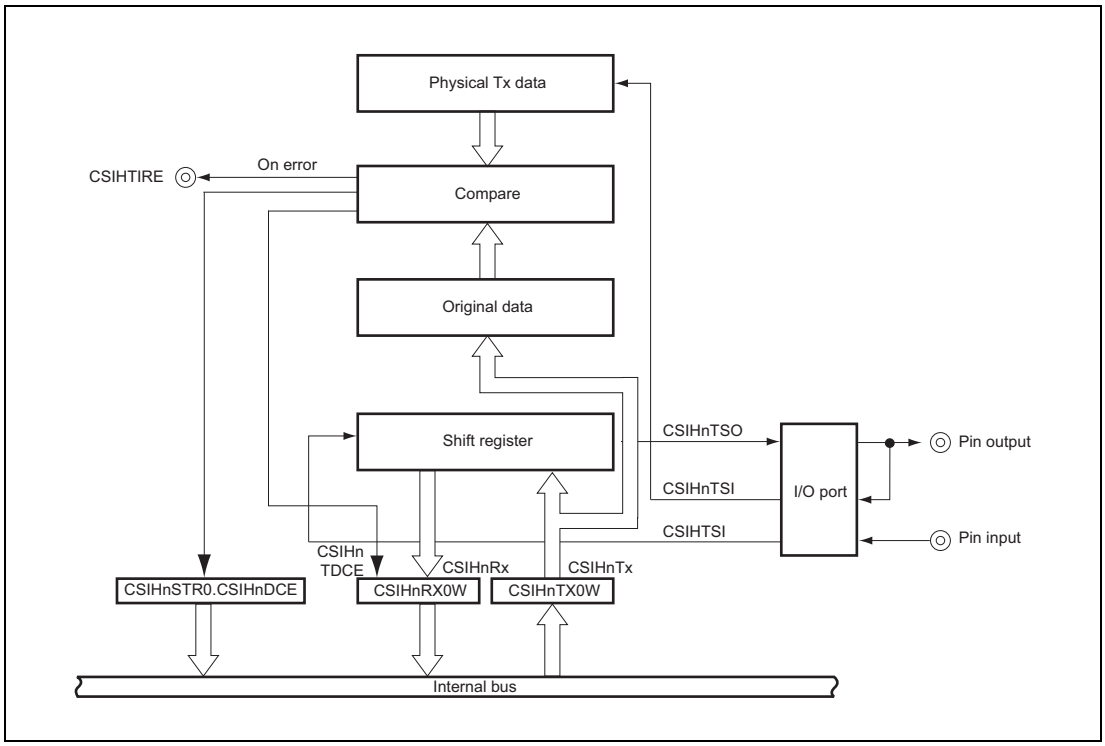


Figure 13.31 Data Consistency Check Functional Block Diagram

(2) Parity check

CSIH can append a parity bit to the last data bit (even if extended data length is used).

The use and type of parity is specified in CSIHnCFGx.CSIHnPSx[1:0].

Parity check is enabled if CSIHnCFGx.CSIHnPSx[1] = 1.

The parity bit is checked after a reception is complete. In case of parity error:

- Interrupt INT_CSIHTIRE is generated.
- The CSIHnSTR0.CSIHnPE bit is set.

Additionally, CSIHnRX0W.CSIHnRPE of data that contains the error is set.

The figure below shows an example.

- Data length is 8 bits.
- The data to be transmitted is 05_H and 35_H.
- Data direction is LSB first.
- Parity type is odd.

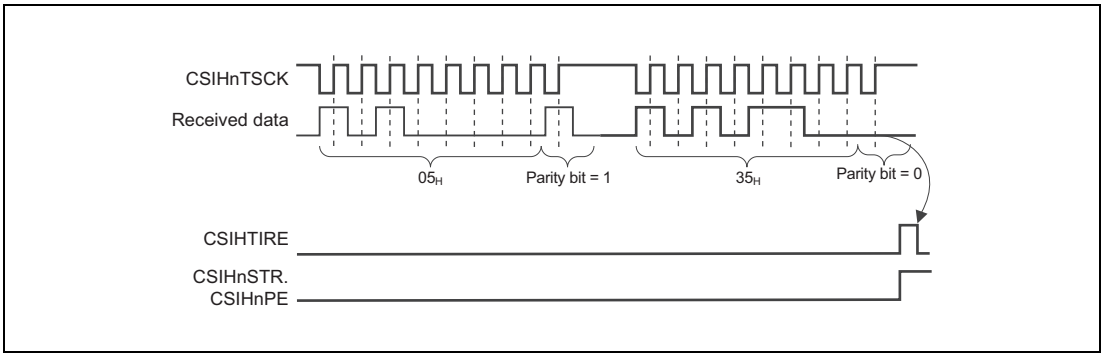


Figure 13.32 Parity Check Example

The parity bit of the first data is 1. There is no parity error, because the total number of ones (including the parity bit) is odd.

The parity bit of the second data is 0. This is detected as a parity error, because the total number of ones (including the parity bit) is even.

If the EDL (extended data length) function is used, the parity bit is added after the last bit of the data.

(3) Overrun error

An overrun error can happen in direct access mode. The overrun error is not generated if data reception is disabled (CSIHnCTL0.CSIHnRXE = 0).

Direct access

In direct access mode, this error occurs when newly received data cannot be transferred from the shift register to the reception register CSIHnRX0W/H. This happens when CSIHnRX0W/H was not read and therefore contains previous reception data.

The following figure illustrates the function.

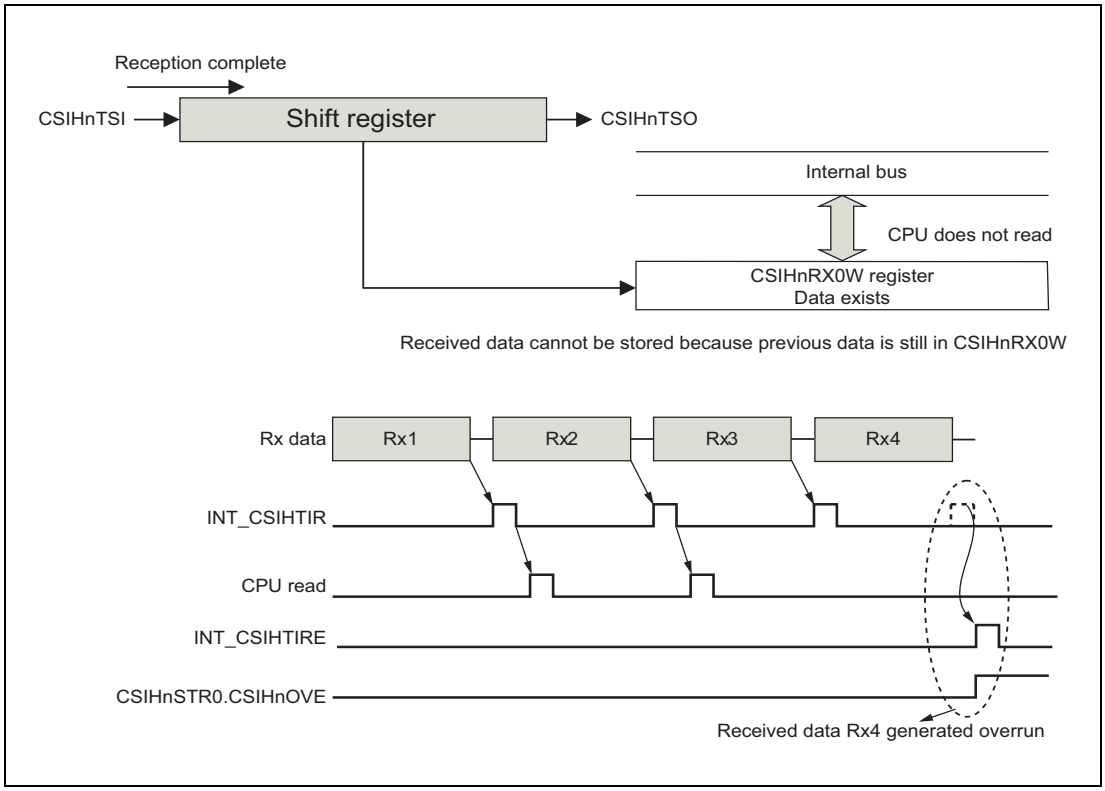


Figure 13.33 Overrun Error Detection in Direct Access

An overrun error causes the following incidents.

- INT_CSIHTIRE interrupt is generated instead of INT_CSIHTIR interrupt
- CSIHnSTR0.CSIHnOVE bit is set
- CSIHnRX0W is rewritten by the received data

NOTE

In slave mode, overrun errors can be avoided by using handshake function. When handshaking is used in slave mode, the receiver (slave) signals the transmitter (master) that the receiver is busy. The transmitter waits until the receiver reads its own register and is ready again.

For details, see **Section 13.5.11, Handshake function**.

13.5.13 Loop-Back Mode

Loop-back mode is a special mode for self-test. This feature is only available in master mode.

When this mode is active (CSIHnCTL1.CSIHnLBM = 1), CSIHnTCSSx is fixed inactive level (the active level is defined by the value of CSIHnCTL1.CSIH0CSLx), and the transmit and receive signals are internally connected, as shown in the figures below. The signals CSIHnTSCK, CSIHnTSO, CSIHnTSI and CSIHnTCSSx are disconnected from the ports. In addition, the CSIHnTSO output level is fixed to low, and CSIHnTSCK is set to the reset level (high) regardless of the value of CSIHnCFGx.CSIHnCKPx register. The rest of CSIH works as in normal operation.

In order to test CSIH, put it in loop-back mode and carry out normal transfer operations. Then check that the received data is the same as the transmitted data.

A loopback test does not affect the destination to which the device is connected.

Table 13.14 Pin Output Level in Loopback Mode

Pin Name	Output Level
CSIHnTSCK(out)	High level
CSIHnTCSSx	Inactive level
CSIHnTSO	Low level (does not depend on the previous values)
Interrupt	Normal function
CSIHnTRY	Normal function (low level)

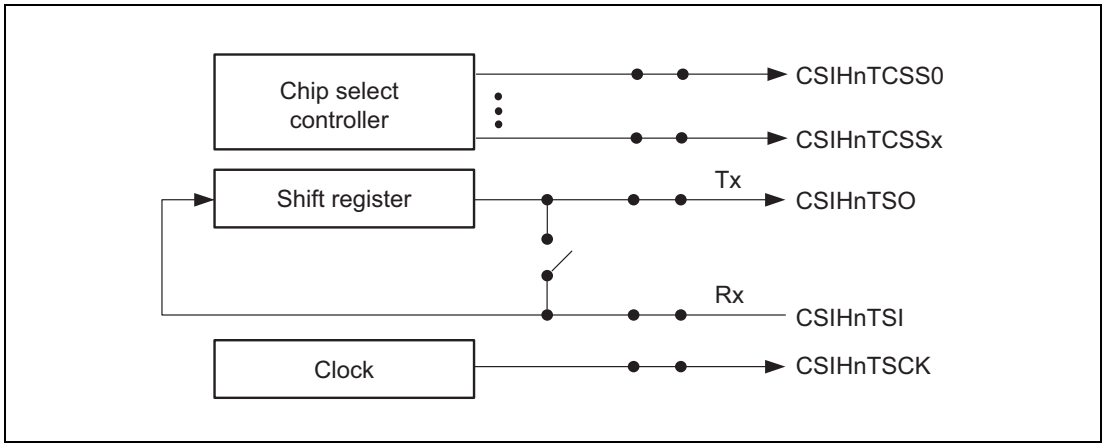


Figure 13.34 Normal Operation

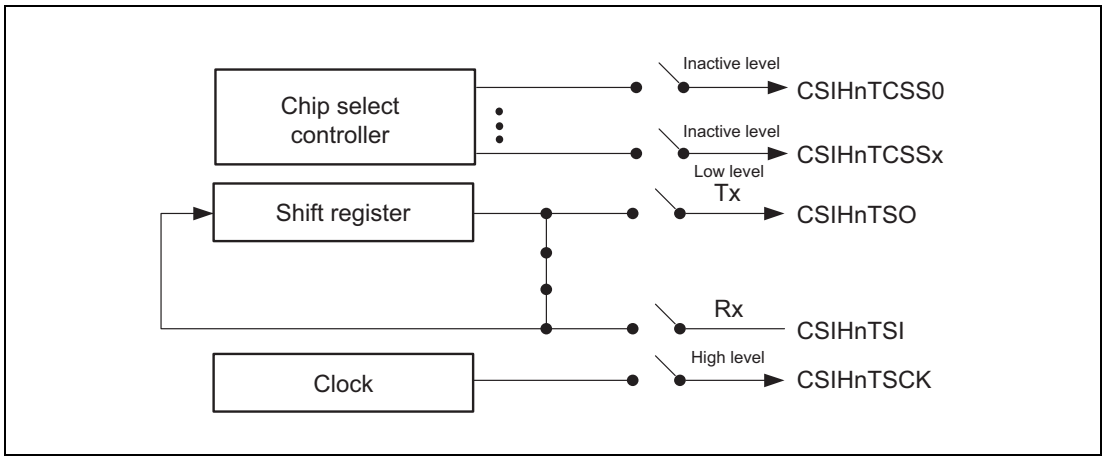


Figure 13.35 Operation in Loop-Back Mode

13.5.14 Enforced Chip Select Idle Setting

The CSIH is able to insert an idle state between the two consecutive transfer data by the setting of CSIHnCFGx.CSIHnIDLx (x: Chip select signal x). Detail is as follows.

1. When CSIHnCFGx.CSIHnIDLx = 0
   If the setting of CSIHnTCSSx for a next transfer data is the same as that for the previous one, an idle state is not inserted and an inter-data time is inserted.
   If the setting of CSIHnTCSSx for a next transfer data is different from the previous one, an idle state is inserted.
2. When CSIHnCFGx.CSIHnIDLx = 1
   An idle state is always inserted even if the setting of CSIHnTCSSx for a next transfer data is the same as that for the previous one.

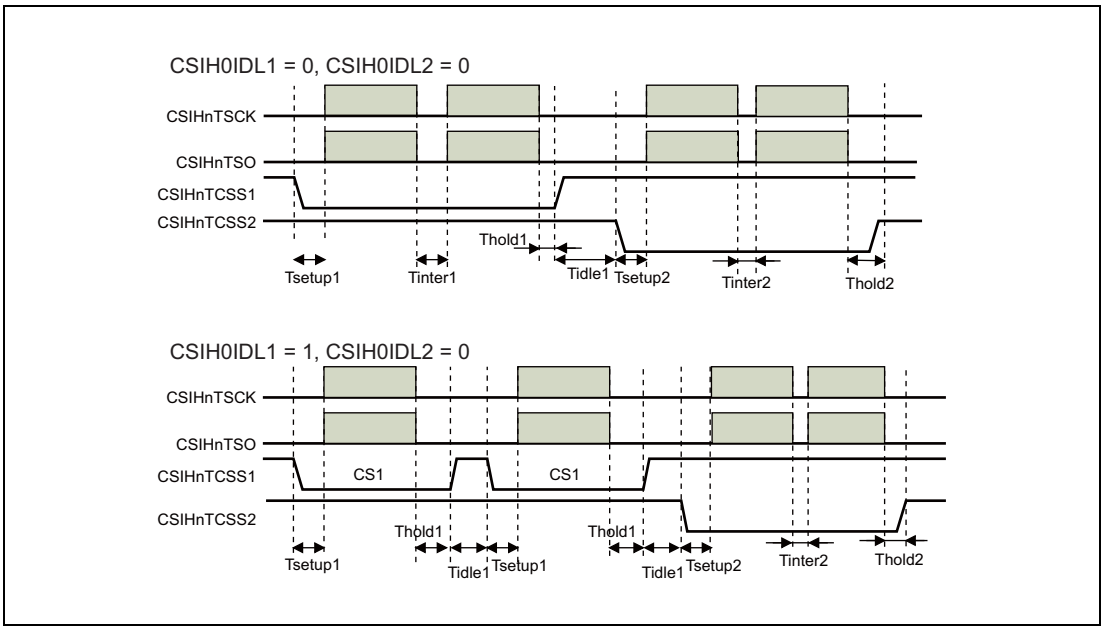


Figure 13.36 Enforced Chip Select Idle Example

13.6 CSIH Control Registers

CSIHn is controlled and operated by means of the registers in the following table.

Table 13.15 CSIH Register Overview

Register name	Shortcut	Address
CSIH control register 0	CSIHnCTL0	<CSIHn_base> + 0000_H
CSIH control register 1	CSIHnCTL1	<CSIHn_base> + 0010_H
CSIH control register 2	CSIHnCTL2	<CSIHn_base> + 0014_H
CSIH status register 0	CSIHnSTR0	<CSIHn_base> + 0004_H
CSIH status clear register 0	CSIHnSTCR0	<CSIHn_base> + 0008_H
CSIH configuration register 0	CSIHnCFG0	<CSIHn_base> + 1044_H
CSIH configuration register 1	CSIHnCFG1	<CSIHn_base> + 1048_H
CSIH configuration register 2	CSIHnCFG2	<CSIHn_base> + $104C_H$
CSIH configuration register 3	CSIHnCFG3	<CSIHn_base> + 1050_H
CSIH configuration register 4	CSIHnCFG4	<CSIHn_base> + 1054_H
CSIH configuration register 5	CSIHnCFG5	<CSIHn_base> + 1058_H
CSIH transmit data register 0 for word access	CSIHnTX0W	<CSIHn_base> + 1008_H
CSIH transmit data register 0 for half word access	CSIHnTX0H	<CSIHn_base> + $100C_H$
CSIH receive data register 0 for word access	CSIHnRX0W	<CSIHn_base> + 1010_H
CSIH receive data register 0 for half word access	CSIHnRX0H	<CSIHn_base> + 1014_H
CSIHn baud rate setting register 0	CSIHnBRS0	<CSIHn_base> + 1068_H
CSIHn baud rate setting register 1	CSIHnBRS1	<CSIHn_base> + $106C_H$
CSIHn baud rate setting register 2	CSIHnBRS2	<CSIHn_base> + 1070_H
CSIHn baud rate setting register 3	CSIHnBRS3	<CSIHn_base> + 1074_H

<CSIHn_base>

The base addresses <CSIHn_base> of the CSIHn are defined in the first part of this section under the key word “Register addresses”.

13.6.1 CSIHnCTL0 — CSIH Control Register 0

This register controls the operation clock and enables/disables transmission/reception and the memory part for transmission and/or reception. It forces the stop of communication at the end of the current job.

Access: This register can be read/written in 8-bit or 1-bit units.

Address: <CSIHn_base> + 0000_H

Value after reset: 00_H This register is initialized by reset operation.

Bit	7	6	5	4	3	2	1	0
	CSIHnPWR	CSIHnTXE	CSIHnRXE	—	—	—	CSIHnJOBE	CSIHnMBS
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R	R	R	R/W	R/W

Table 13.16 CSIHnCTL0 Register Contents

Bit Position	Bit Name	Function
7	CSIHnPWR	Controls the operation clock. 0: Stops operation clock. 1: Provides operation clock. Clearing CSIHnPWR to 0 resets the internal circuits, stops operation, and sets CSIH to standby state. No clock is provided to internal circuits. If CSIHnPWR is cleared during communication, ongoing communication is immediately aborted. In this case, communication setting must be started over.
6	CSIHnTXE	Permits or prohibits transmission. 0: Prohibits transmission. 1: Permits transmission.
5	CSIHnRXE	Permits or prohibits reception. 0: Prohibits reception. 1: Permits reception.
4 to 2	—	Reserved When writing, write the value after reset.
1	CSIHnJOBE	Stops communication at the end of the current job (Communication ends if data is written to the transmission buffer when CSIHnTX0W.CSIHnEOJ = 1 (job completion)): 0: Communication stop is not requested. 1: Stops communication. This bit can be used to abort an ongoing job. This bit is cleared to 0 automatically. If this bit is set to 1, the read value is always 0.
0	CSIHnMBS	Enables direct access mode. 0: CSIH transmit/receive operation is prohibited. 1: Direct access mode Write 1 in the CSIHnMBS bit for transmission/reception via the CSIH.

NOTES

1. Do not modify any of CSIHnTXE, CSIHnRXE, CSIHnJOBE, and CSIHnMBS bits while CSIHnPWR = 0.
   Do not modify the CSIHnMBS bit when CSIHnPWR = 1.
   The CSIHnTXE, CSIHnRXE, and CSIHnMBS bits can be modified when the CSIHnPWR bit is changed from 0 to 1.
   To change the CSIHnMBS bit, do so at the same time as when the CSIHnPWR bit is changed from 0 to 1.
2. Do not modify CSIHnTXE, CSIHnRXE, or CSIHnMBS while data transmission is pending or going on, i.e. if CSIHnSTR0.CSIHnTSF = 1.

13.6.2 CSIHnCTL1 — CSIH Control Register 1

This register specifies the interrupt timing and the interrupt delay mode. It enables/disables extended data length control, data consistency check, loop-back mode, handshake functionality, and job mode. It selects the active output level of each chip select signal and the behavior of the chip select signals after the transfer of the final data.

Access: This register can be read/written in 32-bit units.

Address: <CSIHn_base> + 0010_H

Value after reset: $0000\ 0000_H$ This register is initialized by reset operation.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CSIHn CKR	CSIHn SLIT
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CSIHn CSL7	CSIHn CSL6	CSIHn CSL5	CSIHn CSL4	CSIHn CSL3	CSIHn CSL2	CSIHn CSL1	CSIHn CSL0	CSIHn EDLE	CSIHn JE	CSIHn DCS	CSIHn CSRI	CSIHn LBM	CSIHn SIT	CSIHn HSE	CSIHn SSE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

CAUTION

Changing the contents of this register is only permitted when CSIHnCTL0.CSIHnPWR = 0.

Table 13.17 CSIHnCTL1 Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 18	—	Reserved The write value should be the initial value.
17	CSIHnCKR	CSIHnTSCK Clock Inverting Function 0: Default of CSIHnTSCK is high level. 1: Default of CSIHnTSCK is low level. For details, see **Section 13.6.6, CSIHnCFGx — CSIH Configuration Register x**. **CAUTION** Clear the CSIHnCKPx bit in CSIHnCFGx to 0 when this bit is used without the chip select function.
16	CSIHnSLIT	Selects the timing of interrupt INT_CSIHTIC. 0: Normal interrupt timing (interrupt is generated after the transfer) 1: As soon as the contents of the CSIHnTX0W/H register are transferred to the shift register, an interrupt is generated. For details, see **Section 13.5.10, CSIH interrupt requests**, **(3) INT_CSIHTIC (communication status interrupt)**.
15 to 8	CSIHnCSL[7:0]	Selects the active output level of chip select signal x (CSIHnTCSSx). (n = 0 to 7) 0: Chip select is active low. 1: Chip select is active high. For details, see **Section 13.5.3, Chip Selection (CS) Features**.

Table 13.17 CSIHnCTL1 Register Contents (2/2)

Bit Position	Bit Name	Function
7	CSIHnEDLE	Enables/disables extended data length (EDL) mode. 0: Disables extended data length mode. 1: Enables extended data length mode. For details, see **Section 13.5.7, Data Length Selection**, **(2) Data length greater than 16 bits**.
6	CSIHnJE	Enables/disables job mode. 0: Disables job mode. 1: Enables job mode. For details, see **Section 13.5.3, Chip Selection (CS) Features**, **(3) Job concept**. The CSIHnCTL0.CSIHnJOBE and CSIH0TX0W.CSIH0EOJ bits are enabled only when CSIHnJE = 1. Setting this bit in slave mode is prohibited. **NOTE:** When CSIHnJE = 0, interrupt INTC_CSIHnTIC is generated after one unit of data transmission is completed. When CSIHnJE = 1, interrupt INTC_CSIHnTIC is also generated after one unit of data transmission is completed. In the case of CSIHnTX0W.CSIHnEOJ = 1 and CSIHnCTL0.CSIHnJOBE = 1, however, interrupt INTC_CSIHnTIJC is generated instead of INTC_CSIHnTIC.
5	CSIHnDCS	Enables/disables data consistency check. 0: Disables data consistency check. 1: Enables data consistency check. For details, see **Section 13.5.12, Error Detection**, **(1) Data consistency check**.
4	CSIHnCSRI	Defines chip select signal behavior after last data transfer. 0: Chip select signal holds the active level. 1: Chip select signal returns to the inactive level. The last data is determined at the interrupt timing when CSIHnCTL1.CSIHnSLIT is 1.
3	CSIHnLBM	Controls loop-back mode (LBM). 0: Deactivates loop-back mode. 1: Activates loop-back mode. For details, see **Section 13.5.13, Loop-Back Mode**.
2	CSIHnSIT	Selects interrupt delay mode. 0: No delay is generated. 1: Half clock delay is generated for all interrupts. This bit is only valid in master mode. In slave mode, no delay is generated. For details, see **Section 13.5.10, CSIH interrupt requests**, **(2) Interrupt delay**.
1	CSIHnHSE	Enables/disables handshake function. 0: Disables the handshake function. 1: Enables the handshake function. For details refer to **Section 13.5.11, Handshake function**.
0	CSIHnSSE	Enables/disables the slave select function. 0: Input signal $\overline{\text{CSIHnTSSI}}$ is disabled. 1: Input signal $\overline{\text{CSIHnTSSI}}$ is recognized. If the slave select function is not used, this bit must be set to 0 (see also **Section 13.5.2, Master/Slave Connections**).

Details about CSIHnCTL1.CSIHnSSE are shown in the following tables.

Table 13.18 Operation of the Slave Select Function During Receive

CSIHnCTL0. CSIHnRXE	CSIHnCTL1. CSIHnSSE	$\overline{\text{CSIHnTSSI}}$	Receive operation
0	—	—	Receive is prohibited
1	0	—	Possible
1	1	0	Possible
1	1	1	Disabled

Table 13.19 Operation of the Slave Select Function During Transmission

CSIHnCTL0. CSIHnTXE	CSIHnCTL1. CSIHnSSE	$\overline{\text{CSIHnTSSI}}$	Transmit operation
0	—	—	Transmission is prohibited
1	0	—	Possible
1	1	0	Possible
1	1	1	Disabled

13.6.3 CSIHnCTL2 — CSIH Control Register 2

This register selects operating mode and the basic clock value, and specifies the baud rate.

For details see **Section 13.5.5, Transmission Clock Selection**.

Access: This register can be read/written in 16-bit units.

Address: <CSIHn_base> + 0014_H

Value after reset: $E000_H$ This register is initialized by reset operation.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CSIHnPRS[2:0]			—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R

CAUTION

Changing the contents of this register is only permitted when CSIHnCTL0.CSIHnPWR = 0.

Table 13.20 CSIHnCTL2 Register Contents

Bit Position	Bit Name	Function
15 to 13	CSIHnPRS[2:0]	Selects the basic clock (PRSOUT).
12 to 0	—	Reserved When writing, write the value after reset.

CSIHnPRS2	CSIHnPRS1	CSIHnPRS0	Prescaler Section Bits
0	0	0	PCLK (Master mode)
0	0	1	PCLK/2 (Master mode)
0	1	0	PCLK/4 (Master mode)
0	1	1	PCLK/8 (Master mode)
1	0	0	PCLK/16 (Master mode)
1	0	1	PCLK/32 (Master mode)
1	1	0	PCLK/64 (Master mode)
1	1	1	External clock input from CSIHnTSCK (Slave mode)

CAUTION

Max baud rate configuration is as follows.
- Master mode is PCLK/8.
- Slave mode is PCLK/20.

In master mode, the following bits are used to set the transmission baud rate:

CSIHnCTL2.CSIHnPRS2 to 0, CSIHnCFGx.CSIHnBRSSx1 to 0, CSIHnBRSi.CSIHnBRSi11 to 0

In addition, any of the four different baud rate settings that are specified by the CSIHnBRSi (i = 0 to 3).CSIHnBRSi11 to 0 (i = 0 to 3) bit is selected according to the chip select signal. To select the baud rate setting for each chip select signal, use the CSIHnCFGx.CSIHnBRSSx1 to 0 bit.

The following table shows the relationship between CSIHnCFGx.CSIHnBRSSx1 to 0 and CSIHnBRSi (i = 0 to 3).CSIHnBRSi11 to 0 (i = 0 to 3).

CSIHnCFGx CSIHnBRSSx1 to 0	Baud Rate Setting Bits to be Selected
00	CSIHnBRS0.CSIH0BRS0 11 to 0
01	CSIHnBRS1.CSIH0BRS1 11 to 0
10	CSIHnBRS2.CSIH0BRS2 11 to 0
11	CSIHnBRS3.CSIH0BRS3 11 to 0

When the value of CSIHnPRS[2:0] is m, the following table shows the relationship between the baud rate setting (CSIHnBRSi[11:0]) and the baud rate selected by the CSIHnBRSSx[1:0] bits.

CSIHnBRSx11 to 0 (k)	Baud Rate Selection Bits
0	BRG stopped
1	PLCK / (2^m × 1 × 2)
2	PLCK / (2^m × 2 × 2)
3	PLCK / (2^m × 3 × 2)
4	PLCK / (2^m × 4 × 2)
:	:
k	PLCK / (2m × k × 2)
:	:
4095	PLCK / (2^m × 4095 × 2)

13.6.4 CSIHnSTR0 — CSIH Status Register 0

This register indicates the status of CSIH.

Access: This register can be read in 32-bit units.

Address: <CSIHn_base> + 0004_H

Value after reset: $0000\ 0010_H$ This register is initialized by reset operation.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	CSIHn TSF	—	—	—	CSIHn DCE	—	CSIHn PE	CSIHn OVE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 13.21 CSIHnSTR0 Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 8	—	Reserved
7	CSIHnTSF	Transfer status flag 0: Idle state 1: Communication is in progress or being prepared. The timing to set or clear this bit is as follows. **Master mode** / **Set by** / **Cleared by** Transmit-only mode, Transmit/receive mode, Receive-only mode: Data is written to a transmit register / Within a half clock of the last serial clock edge **Slave mode** / **Set by** / **Cleared by** Transmit-only mode, Transmit/receive mode: Writing to a transmit register / Within a half clock of the last serial clock edge Receive-only mode: Input timing of CSIHnTSCK
6 to 4	—	Reserved
3	CSIHnDCE	Data consistency check error flag 0: No data consistency error is detected. 1: Data consistency error is detected. This bit is cleared by writing 1 to CSIHnSTCR0.CSIHnDCEC. When this bit is set to 1 and cleared to 0 simultaneously, setting to 1 is prioritized. This bit is initialized when CSIHnCTL0.CSIHnPWR changes from 0 to 1 or from 1 to 0.
2	—	Reserved
1	CSIHnPE	Parity error flag 0: No parity error is detected. 1: Parity error is detected. For details, see **(2) Parity check** in **Section 13.5.12, Error Detection**. This bit is cleared by the CSIHnPEC bit in CSIHnSTCR0. When this bit is set to 1 and cleared to 0 simultaneously, clearing to 0 is prioritized.

Table 13.21 CSIHnSTR0 Register Contents (2/2)

Bit Position	Bit Name	Function
0	CSIHnOVE	Overrun error flag 0: No overrun error is detected. 1: Overrun error is detected. This bit is cleared by writing 1 to CSIHnSTCR0.CSIHnOVEC. When this bit is set to 1 and cleared to 0 simultaneously, setting to 1 is prioritized. This bit is Initialized when CSIHnCTL0.CSIHnPWR changes from 0 to 1 or from 1 to 0.

13.6.5 CSIHnSTCR0 — CSIH Status Clear Register 0

This register clears the status flags of the CSIHnSTR0 status register.

Access: This register can be read/written in 16-bit units.
When read, the value 0000_H is always returned.

Address: <CSIHn_base> + 0008_H

Value after reset: 0000_H This register is initialized by reset operation.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	CSIHn DCEC	—	CSIHn PEC	CSIHn OVEC
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R/W	R/W

Table 13.22 CSIHnSTCR0 Register Contents

Bit Position	Bit Name	Function
15 to 4	—	Reserved When writing, write the value after reset.
3	CSIHnDCEC	Controls the data consistency error flag clear command. 0: No operation. The read value is always 0. 1: Clears the data consistency error flag (CSIHnSTR0.CSIHnDCE).
2	—	Reserved When writing, write the value after reset.
1	CSIHnPEC	Controls the parity error flag clear command. 0: No operation. The read value is always 0. 1: Clears the parity error flag (CSIHnSTR0.CSIHnPE).
0	CSIHnOVEC	Controls the overrun error flag clear command. 0: No operation. The read value is always 0. 1: Clears the overrun error flag (CSIHnSTR0.CSIHnOVE).

13.6.6 CSIHnCFGx — CSIH Configuration Register x

These registers specify for each chip select signal CSIHnTCSSx prescaler, parity, data length, recessive configuration for broadcasting, serial data direction, clock and data phase, idle enforcement configuration, idle time, hold time, inter-data time and setup time.

Slave mode

In slave mode, the transmission protocol setting of the CSIHnCFG0 register is effective.

- CSIHnPS0[1:0]: parity usage
- CSIHnDLS0[3:0]: data length selection
- CSIHnDIR0: data direction
- CSIHnCKP0, CSIHnDAP0: clock and data phase

The settings of the registers other than CSIHnCFG0 are ignored in slave mode.

Access: This register can be read/written in 32-bit units.

Address: CSIHnCFG0: <CSIHn_base> + 1044_H
CSIHnCFG1: <CSIHn_base> + 1048_H
CSIHnCFG2: <CSIHn_base> + $104C_H$
CSIHnCFG3: <CSIHn_base> + 1050_H
CSIHnCFG4: <CSIHn_base> + 1054_H
CSIHnCFG5: <CSIHn_base> + 1058_H

Value after reset: $0000\ 0000_H$ This register is initialized by reset operation.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CSIHn BRSSx[1:0]		CSIHn PSx[1:0]		CSIHnDLSx[3:0]				—	—	—	—	CSIHn RCBx	CSIHn DIRx	CSIHn CKPx	CSIHn DAPx
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CSIHn IDLx	CSIHnIDx[2:0]			CSIHnHDx[3:0]				CSIHnINx[3:0]				CSIHnSPx[3:0]			
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

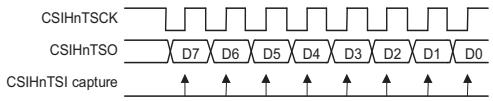
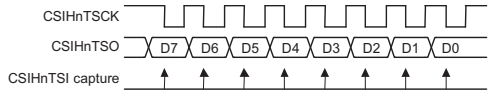
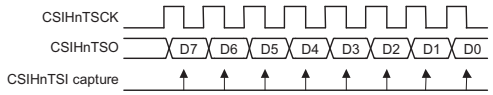
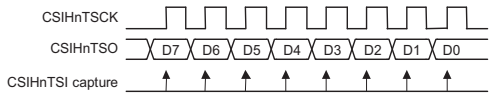
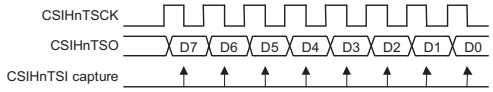
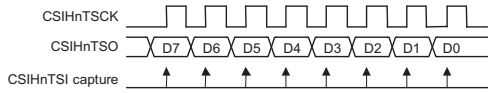
CAUTION

Rewrite is possible only when CSIHnCTL0.CSIHnPWR = 0 (for write access with the same value, rewrite is possible when CSIHnCTL0.CSIHnPWR = 1).

Table 13.23 CSIHnCFGx Register Contents (1/4)

<table>
<tr><th>Bit Position</th><th>Bit Name</th><th>Function</th></tr>
<tr><td>31, 30</td><td>CSIHnBRSSx [1:0]</td><td>A register used to set the baud rate.
<table>
<tr><th>CSIHn BRSSx1</th><th>CSIHn BRSSx0</th><th>Baud rate setting register selection</th></tr>
<tr><td>0</td><td>0</td><td>The baud rate is set according to the CSIH0BRS0 setting</td></tr>
<tr><td>0</td><td>1</td><td>The baud rate is set according to the CSIH0BRS1 setting</td></tr>
<tr><td>1</td><td>0</td><td>The baud rate is set according to the CSIH0BRS2 setting</td></tr>
<tr><td>1</td><td>1</td><td>The baud rate is set according to the CSIH0BRS3 setting</td></tr>
</table>
The maximum value for setting the baud rate, combining the CSIH0CTL2.PRS2 to 0 setting, must be as follows:
Master mode: PLCK/8
Slave mode: PLCK/20</td></tr>
<tr><td>29, 28</td><td>CSIHnPSx[1:0]</td><td>Selects the parity for sending or receiving chip select signal x.
<table>
<tr><th>CSIHn PSx1</th><th>CSIHn PSx0</th><th>Transmission</th><th>Receive</th></tr>
<tr><td>0</td><td>0</td><td>No parity is transmitted</td><td>No parity is expected</td></tr>
<tr><td>0</td><td>1</td><td>Adds parity bit fixed to 0</td><td>Parity bit is expected but not judged</td></tr>
<tr><td>1</td><td>0</td><td>Adds odd parity only</td><td>Odd parity bit is expected</td></tr>
<tr><td>1</td><td>1</td><td>Adds even parity</td><td>Even parity bit is expected</td></tr>
</table></td></tr>
<tr><td>27 to 24</td><td>CSIHnDLSx [3:0]</td><td>Selects the data length for chip select signal x.
<table>
<tr><th>CSHBAnDLSx[3:0]</th><th>Data length</th></tr>
<tr><td>0000_B</td><td>16 bits</td></tr>
<tr><td>0001_B</td><td>1 bit</td></tr>
<tr><td>0010_B</td><td>2 bits</td></tr>
<tr><td>:</td><td>:</td></tr>
<tr><td>1111_B</td><td>15 bits</td></tr>
</table>
CAUTION
When CSIH0TX0W.CSIH0EDL = 1, setting these bits has no effect (the data length is 16 bits).
When CSIH0TX0W.CSIH0EDL = 0, the settings of these bits are effective.
Setting 1 bit is only possible when the data length of the previous data is 16 bits as specified by CSIH0EDL = 1.</td></tr>
<tr><td>23 to 20</td><td>—</td><td>Reserved
When writing, write the value after reset.</td></tr>
<tr><td>19</td><td>CSIHnRCBx</td><td>Selects the recessive configuration for broadcasting for chip select x:
0: Dominant (higher priority)
1: Recessive (lower priority)
For details, see Section 13.5.3, Chip Selection (CS) Features, (1) Configuration registers.</td></tr>
<tr><td>18</td><td>CSIHnDIRx</td><td>Selects the serial data direction.
0: Data is sent/received with MSB first.
1: Data is sent/received with LSB first.
For details see Section 13.5.8, Serial Data Direction Selection.</td></tr>
</table>

Table 13.23 CSIHnCFGx Register Contents (2/4)

<table>
<tr><th>Bit Position</th><th>Bit Name</th><th>Function</th></tr>
<tr><td>17</td><td>CSIHnCKPx</td><td rowspan="2">CSIHnCKPx: Clock phase selection bit
CSIHnDAPx: Data phase selection bit
• CSIHnCTL1.CSIHnCKR = 0
<table><tr><th>CSIHn CKPx</th><th>CSIHn DAPx</th><th>Clock and Data Phase Selection</th></tr><tr><td>0</td><td>0</td><td>  </td></tr><tr><td>0</td><td>1</td><td>  </td></tr><tr><td>1</td><td>0</td><td>  </td></tr><tr><td>1</td><td>1</td><td>  </td></tr></table>• CSIHnCTL1.CSIHnCKR = 1
<table><tr><th>CSIHn CKPx</th><th>CSIHn DAPx</th><th>Clock and Data Phase Selection</th></tr><tr><td>0</td><td>0</td><td>  </td></tr><tr><td>0</td><td>1</td><td>  </td></tr><tr><td>1</td><td>x</td><td>Setting prohibited</td></tr></table></td></tr>
<tr><td>16</td><td>CSIHnDAPx</td></tr>
<tr><td>15</td><td>CSIHnIDLx</td><td>Selects the idle enforcement configuration for chip select x:
0: An idle state is always inserted between two consecutive transfer data when the setting of the CSIHnCSx bit in CSIHnTX0W for the latter is different from that of the former. Meanwhile, an idle state is not inserted between two consecutive transfer data when the settings of the CSIHnCSx bit in CSIHnTX0W for the latter is the same as that of the former.
1: Regardless of the settings of the CSIHnCSx bit in CSIHnTX0W for two consecutive transfer data, an idle state is always inserted between the two.

This bit is only available in master mode.
For details about the enforced idle state, see Section 13.5.14, Enforced Chip Select Idle Setting.</td></tr>
</table>

Table 13.23 CSIHnCFGx Register Contents (3/4)

Bit Position	Bit Name	Function
14 to 12	CSIHnIDx[2:0]	Selects the idle time for chip select signal x. These bits are only available in master mode.
11 to 8	CSIHnHDx[3:0]	Specifies the hold time for chip select signal x in transmission clock cycles. These bits are only available in master mode.

CSIHn IDx[2:0]	Idle time
000_B	0.5 transmission clock cycles
001_B	1.0 transmission clock cycle
010_B	1.5 transmission clock cycles
011_B	2.5 transmission clock cycles
100_B	3.5 transmission clock cycles
101_B	4.5 transmission clock cycles
110_B	6.5 transmission clock cycles
111_B	8.5 transmission clock cycles

CSIHn HDx[3:0]	Hold time when CSIHnCTL1.CSIHnSIT = 0	Hold time when CSIHnCTL1.CSIHnSIT = 1
0000_B	0.5 transmission clock cycles	1.0 transmission clock cycle
0001_B	1.0 transmission clock cycle	1.5 transmission clock cycles
0010_B	1.5 transmission clock cycles	2.0 transmission clock cycles
0011_B	2.5 transmission clock cycles	3.0 transmission clock cycles
0100_B	3.5 transmission clock cycles	4.0 transmission clock cycles
0101_B	4.5 transmission clock cycles	5.0 transmission clock cycles
0110_B	6.5 transmission clock cycles	7.0 transmission clock cycles
0111_B	8.5 transmission clock cycles	9.0 transmission clock cycles
1000_B	9.5 transmission clock cycles	10.0 transmission clock cycles
1001_B	10.5 transmission clock cycles	11.0 transmission clock cycles
1010_B	11.5 transmission clock cycles	12.0 transmission clock cycles
1011_B	12.5 transmission clock cycles	13.0 transmission clock cycles
1100_B	14.5 transmission clock cycles	15.0 transmission clock cycles
1101_B	16.5 transmission clock cycles	17.0 transmission clock cycles
1110_B	18.5 transmission clock cycles	19.0 transmission clock cycles
1111_B	20.5 transmission clock cycles	21.0 transmission clock cycles

Table 13.23 CSIHnCFGx Register Contents (4/4)

Bit Position	Bit Name	Function
7 to 4	CSIHnINx[3:0]	Specifies the inter-data delay time for chip select signal x in transmission clock cycles.

CSIHn INx[3:0]	Inter-Data time when CSIHnCTL1.CSIHnSIT = 0	Inter-Data time when CSIHnCTL1.CSIHnSIT = 1
0000_B	0.0 transmission clock cycles	0.5 transmission clock cycles
0001_B	0.5 transmission clock cycles	1.0 transmission clock cycle
0010_B	1.0 transmission clock cycle	1.5 transmission clock cycles
0011_B	2.0 transmission clock cycles	2.5 transmission clock cycles
0100_B	3.0 transmission clock cycles	3.5 transmission clock cycles
0101_B	4.0 transmission clock cycles	4.5 transmission clock cycles
0110_B	6.0 transmission clock cycles	6.5 transmission clock cycles
0111_B	8.0 transmission clock cycles	8.5 transmission clock cycles
1000_B	9.0 transmission clock cycles	9.5 transmission clock cycles
1001_B	10.0 transmission clock cycles	10.5 transmission clock cycles
1010_B	11.0 transmission clock cycles	11.5 transmission clock cycles
1011_B	12.0 transmission clock cycles	12.5 transmission clock cycles
1100_B	14.0 transmission clock cycles	14.5 transmission clock cycles
1101_B	16.0 transmission clock cycles	16.5 transmission clock cycles
1110_B	18.0 transmission clock cycles	18.5 transmission clock cycles
1111_B	20.0 transmission clock cycles	20.5 transmission clock cycles

These bits are only available in master mode.

3 to 0	CSIHnSPx[3:0]	Specifies the setup time for chip select signal x in transmission clock cycles.

CSIHn SPx[3:0]	Setup time
0000_B	0.5 transmission clock cycles
0001_B	1.0 transmission clock cycle
0010_B	1.5 transmission clock cycles
0011_B	2.5 transmission clock cycles
0100_B	3.5 transmission clock cycles
0101_B	4.5 transmission clock cycles
0110_B	6.5 transmission clock cycles
0111_B	8.5 transmission clock cycles
1000_B	9.5 transmission clock cycles
1001_B	10.5 transmission clock cycles
1010_B	11.5 transmission clock cycles
1011_B	12.5 transmission clock cycles
1100_B	14.5 transmission clock cycles
1101_B	16.5 transmission clock cycles
1110_B	18.5 transmission clock cycles
1111_B	20.5 transmission clock cycles

These bits are only available in master mode.

13.6.7 CSIHnTX0W — CSIH Transmit Data Register 0 for Word Access

This register stores transmission data. In addition, it specifies the communication interrupt request, the end-of-job, the extended data length, and the chip select activation.

Access: This register can be read/written in 32-bit units.

Address: <CSIHn_base> + 1008_H

Value after reset: Undefined

CAUTIONS

When CSIH0CTL0.CSIH0TXE = CSIH0CTL0.CSIH0RXE = 0 in direct access mode, write access to this register is prohibited.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	CSIHnEOJ	CSIHnEDL	—	—	—	—	—	CSIHnCS7	CSIHnCS6	CSIHnCS5	CSIHnCS4	CSIHnCS3	CSIHnCS2	CSIHnCS1	CSIHnCS0
Value after reset	0			0	0	0	0	0								
R/W	R	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CSIHnTX[15:0]															
Value after reset																
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 13.24 CSIHnTX0W Register Contents (1/2)

Bit position	Bit name	Function
31	—	Reserved When writing, write the value after reset.
30	CSIHnEOJ	Specifies the end of a job. 0: Indicates that the job does not end. The job continues. 1: Indicates end-of-job data. **CAUTION** This bit is only valid when job mode is enabled (CSIHnCTL1.CSIHnJE = 1). This bit must be set to 0 in slave mode.
29	CSIHnEDL	Specifies whether the associated data requires the extended data length (EDL) option. 0: Normal operation 1: Enables the extended data length. The associated data is transmitted as a 16-bit packet. No inter-data time or idle time will be inserted after the data is transmitted. If CSIHnCTL1.CSIHnEDLE = 1 and CSIHnTX0W.CSIHnEDL = 1, the subsequent data must have the same CS selection. If CS is modified for the subsequent data, the correct operation is not assured. **CAUTION** This bit is only available if CSIHnCTL1.CSIHnEDLE = 1.
28 to 24	—	Reserved When writing, write the value after reset.

Table 13.24 CSIHnTX0W Register Contents (2/2)

Bit position	Bit name	Function
23 to 16	CSIHnCSx	Activates one or several chip select signals. 0: Activates chip select x for the associated transmission. 1: Deactivates chip select x for the associated transmission. Setting CSIHnTX0W.CSIHnCS[7:0] = FF_H is prohibited. **CAUTION** If several chip select signals are enabled for broadcasting, the configuration of one with CSIHnCFGx.CSIHnRCBx = 0 (dominant) is used. In this case, all dominant chip select signals must be set to precisely the same value. In slave mode, set the CSIHnCSx bit to FE_H.
15 to 0	CSIHnTX[15:0]	Stores the transmission data.

13.6.8 CSIHnTX0H — CSIH Transmit Data Register 0 for Half Word Access

This register stores the transmission data. This register is the same as bits 15 to 0 of register CSIHnTX0W.

Access: This register can be read/written in 16-bit units.

Address: <CSIHn_base> + $100C_H$

Value after reset: Undefined

CAUTIONS

When CSIH0CTL0.CSIH0TXE = CSIH0CTL0.CSIH0RXE = 0 in direct access mode, write access to this register is prohibited.

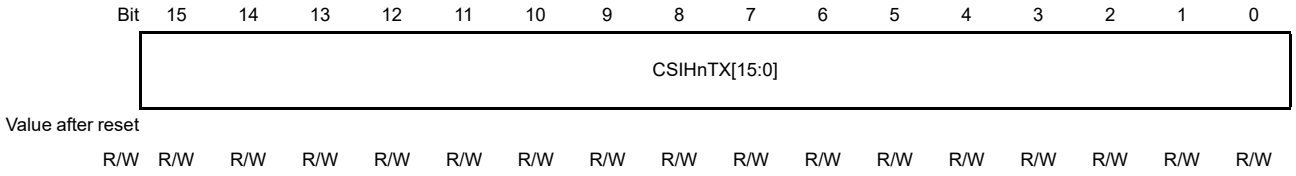


Table 13.25 CSIHnTX0H Register Contents

Bit position	Bit name	Function
15 to 0	CSIHnTX[15:0]	Stores the transmission data.

13.6.9 CSIHnRX0W — CSIH Receive Data Register 0 for Word Access

This register stores the received data.

Access: This register can be read in 32-bit units.

Address: <CSIHn_base> + 1010_H

Value after reset: Undefined

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	CSIHnRPE	CSIHnTDCE	CSIHnCS7	CSIHnCS6	CSIHnCS5	CSIHnCS4	CSIHnCS3	CSIHnCS2	CSIHnCS1	CSIHnCS0
Value after reset	0	0	0	0	0	0	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CSIHnRX[15:0]															
Value after reset	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

CAUTION

This register can be read in direct access mode and transmit-only buffer mode while CSIHnCTL0.CSIHnPWR = 1.
This register is initialized when CSIHnCTL0.CSIHnPWR changes from 0 to 1 or from 1 to 0.

Table 13.26 CSIHnRX0W Register Contents

Bit position	Bit name	Function
31 to 26	—	Reserved
25	CSIHnRPE	Indicates whether a reception data parity error was detected. 0: No parity error was detected on the associated reception data. 1: A parity error was detected on the associated reception data.
24	CSIHnTDCE	Indicates whether a transmission data consistency error was detected. 0: No consistency error was detected on the associated transmission data. 1: A consistency error was detected on the associated transmission data.
23 to 16	CSIHnCSx	Indicates which chip select signal was activated. 0: Chip select x was activated for the associated reception. 1: Chip select x was deactivated for the associated reception.
15 to 0	CSIHnRX[15:0]	Stores the received data.

13.6.10 CSIHnRX0H — CSIH Receive Data Register 0 for Half Word Access

This register stores the received data. This register is the same as bits 15 to 0 of register CSIHnRX0W.

Access: This register can only be read in 16-bit units.

Address: <CSIHn_base> + 1014_H

Value after reset Undefined

CAUTIONS

This register can be read in direct access mode and transmit-only buffer mode while CSIHnCTL0.CSIHnPWR = 1.

This register is initialized when CSIHnCTL0.CSIHnPWR changes from 0 to 1 or from 1 to 0.

Table 13.27 CSIHnRX0H Register Contents

Bit position	Bit name	Function
15 to 0	CSIHnRX[15:0]	Stores the received data.

13.6.11 CSIHnBRSi — CSIHn Baud Rate Setting Register i

This register sets the baud rate for each chip select signal.

CSIHnCFGx.CSIHnBRSSx1-0 bits, one of the four types of baud rate settings can be selected for each chip select signal.

Access: This register can be read/written in 16-bit units.

Address: CSIHnBRS0: <CSIHn_base> + 1068_H
CSIHnBRS1: <CSIHn_base> + $106C_H$
CSIHnBRS2: <CSIHn_base> + 1070_H
CSIHnBRS3: <CSIHn_base> + 1074_H

Value after reset: 0000_H. This register is initialized by reset operation.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	CSIHnBRSi[11:0]											
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 13.28 CSIHnBRSi Register Contents

Bit Position	Bit Name	Function
15 to 12	—	Reserved The write value should be the initial value.
11 to 0	CSIHnBRSi [11:0]	0: BRG stopped 1: PCLK / (2^m × 1 × 2) 2: PCLK / (2^m × 2 × 2) 3: PCLK / (2^m × 3 × 2) 4: PCLK / (2^m × 4 × 2) : k: PCLK / (2^m × k × 2) : 4095: PCLK / (2^m × 4095 × 2) m is the value of CSIHnCTL2.CSIHnPRS[2:0].

13.7 Operating Procedures

13.7.1 Procedures in Direct Access Mode

Two examples for a master are provided, one with job mode disabled, and the other one with job mode enabled.

(1) Transmission/reception in master mode when job mode is disabled

The procedures below are based on the assumption that:

- The transmission data length is 8 bits (CSIHnCFGx.CSIHnDLSx[3:0] = $1000_{B)}$.
- Transmission direction is MSB first (CSIHnCFGx.CSIHnDIRx = 0).
- Normal clock and data phase (CSIHnCFGx.CSIHnCKPx = 0, CSIHnCFGx.CSIHnDAPx = 0)
- No interrupt delay (CSIHnCTL1.CSIHnSIT = 0)
- Job mode is disabled (CSIHnCTL1.CSIHnJE = 0).
- Normal INT_CSIHTIC interrupt timing (CSIHnCTL1.CSIHnSLIT = 0)
- Direct access mode (CSIHnCTL0.CSIHnMBS = 1)

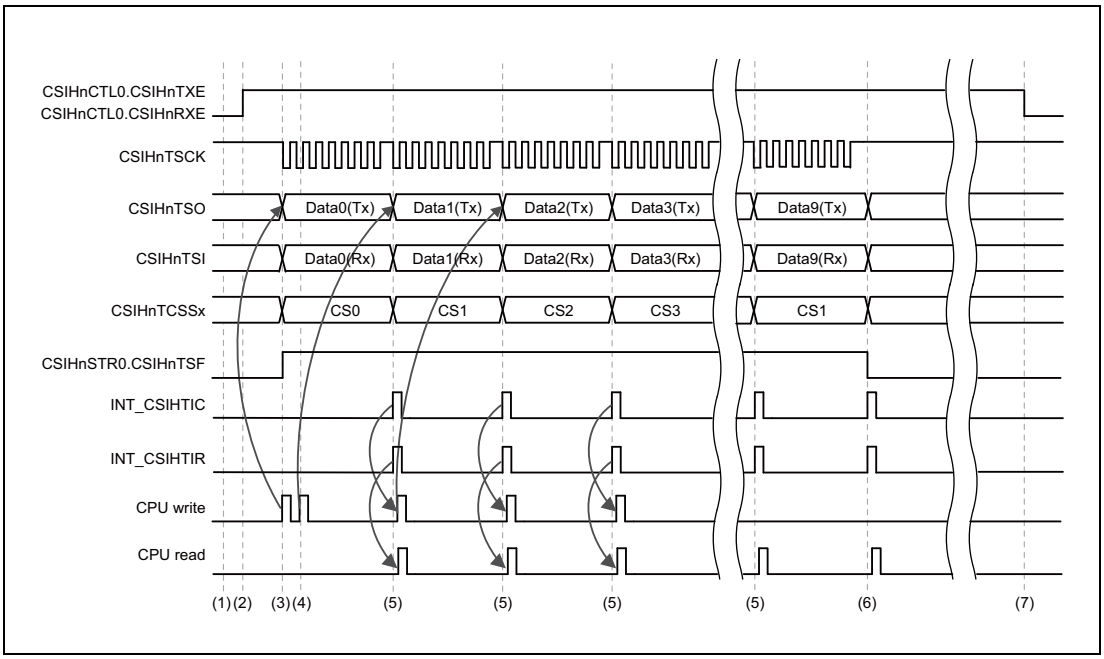


Figure 13.37 Master in Direct Access Mode, CSIHnCTL1.CSIHnJE = 0

Procedure:

(1) Configure the communication protocol in the CSIHnCFGx register. This example uses chip select signals CS0 to CS3.
Specify the transfer mode and job mode by setting the corresponding bits in registers CSIHnCTL1 and CSIHnCTL2.

(2) In the CSIHnCTL0 register, set CSIHnPWR = 1 (enables the clock), CSIHnTXE = 1 (permits transmission), CSIHnRXE = 1 (permits reception), and CSIHnMBS = 1 (direct access mode).

(3) Write the first data to be sent to the transmission register, CSIHnTX0W. Within the same write operation, activate CS0. Transmission starts automatically when the first data becomes available.

(4) Write the second data to CSIHnTX0W. If required, you can change the CS to address a different device. Writing the second data immediately after the first one avoids unnecessary delays between the data.

(5) After every transmission/reception of a data, the interrupts INT_CSIHTIC and INT_CSIHTIR are generated:
- INT_CSIHTIC indicates that the next data can be written to CSIHnTX0W.
- INT_CSIHTIR indicates that the reception register, CSIHnRX0W must be read.

(6) No more write action is required after completion of data 8. Data 9 (the last packet) has been written in advance.
However, reception register CSIHnRX0W must be read after completion of writing data 8 and 9.

(7) Finally, clear CSIHnCTL0.CSIHnTXE and CSIHnCTL0.CSIHnRXE to prohibit the transmit/receive operation. When no communication is conducted, set CSIHnCTL0.CSIHnPWR to 0 to minimize power consumption of the CSIHn.

(2) Transmission/reception in master mode when job mode is enabled

The procedure below is based on the assumption that:

- The transmission data length is 8 bits (CSIHnCFGx.CSIHnDLSx[3:0] = 1000_B).
- Transmission direction is MSB first (CSIHnCFGx.CSIHnDIRx = 0).
- Normal clock and data phase (CSIHnCFGx.CSIHnCKPx = 0, CSIHnCFGx.CSIHnDAPx = 0)
- No interrupt delay (CSIHnCTL1.CSIHnSIT = 0)
- Job mode is enabled (CSIHnCTL1.CSIHnJE = 1).
- Normal INT_CSIHTIC interrupt timing (CSIHnCTL1.CSIHnSLIT = 0)
- Direct access mode (CSIHnCTL0.CSIHnMBS = 1)
- Two jobs, each of them sends three data.

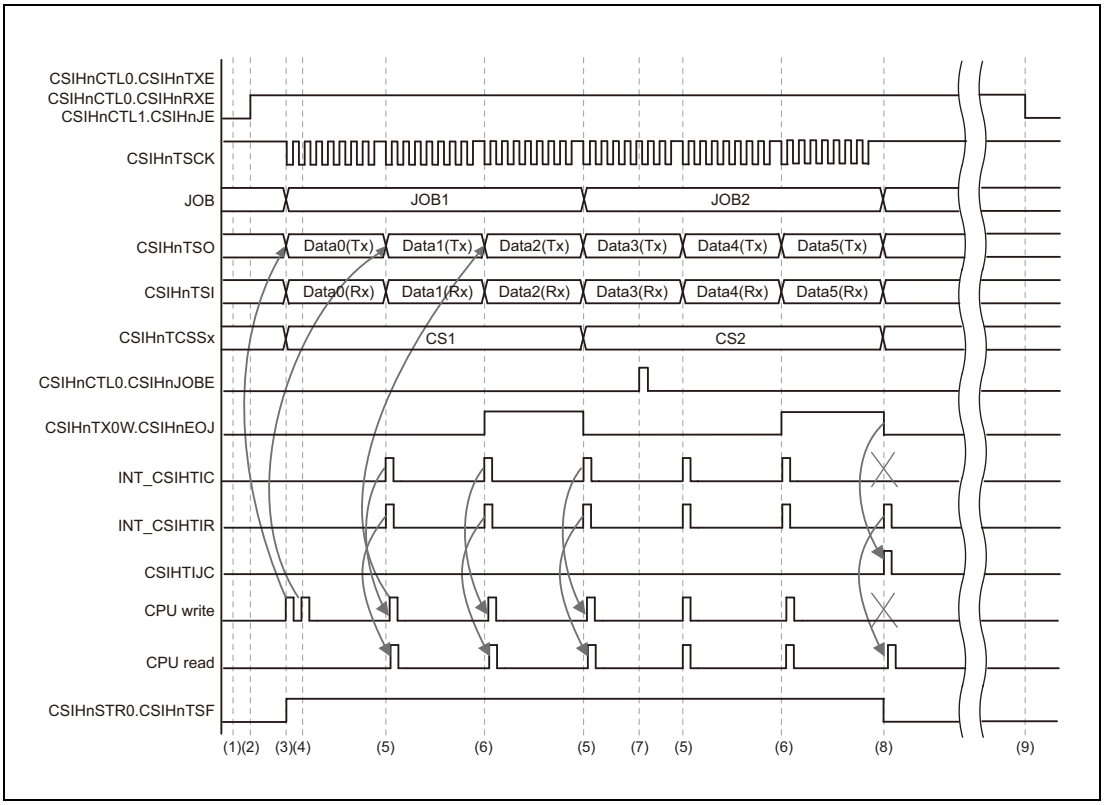


Figure 13.38 Master in Direct Access Mode, CSIHnCTL1.CSIHnJE = 1

Procedure:

(1) Configure the communication protocol in the CSIHnCFGx register. This example uses chip select signals CS1 and CS2.
Specify the transfer mode and job mode by setting the corresponding bits in registers CSIHnCTL1 and CSIHnCTL2.

(2) In the CSIHnCTL0 register, set CSIHnPWR = 1 (enables the clock), CSIHnTXE = 1 (permits transmission), CSIHnRXE = 1 (permits the reception), and CSIHnMBS = 1 (selects direct access mode).

(3) Write the first data to be sent to the transmission register CSIHnTX0W. Transmission starts automatically when the first data becomes available.
The CSIHnSTR0.CSIHnTSF flag indicates that communication is in progress.

(4) Write the second data to CSIHnTX0W. Writing the second data immediately after the first one avoids unnecessary delays between the data.

(5) After every data transmission/reception, the interrupt requests, INT_CSIHTIC and INT_CSIHTIR are generated.
- INT_CSIHTIC indicates that the next data can be written to CSIHnTX0W.
- INT_CSIHTIR indicates that the reception register, CSIHnRX0W must be read.

(6) Setting CSIHnTX0W.CSIHnEOJ = 1 indicates that the last data of the current job is sent. After that, the next job may begin.

(7) By setting CSIHnCTL0.CSIHnJOBE = 1, communication is forced to stop at the end of the current job (JOB2).

(8) After the forced stop of communication, the interrupt request, INT_CSIHTIC is replaced by INT_CSIHTIJC. INT_CSIHTIR is generated as usual.
The interrupt request, INT_CSIHTIJC indicates a forced stop of communication at the end of the current job.
The interrupt request, INT_CSIHTIC is not generated. Additionally, the transmission data available in the CSIHnTX0W register is not sent.

(9) Finally, clear CSIHnCTL0.CSIHnTXE and CSIHnCTL0.CSIHnRXE to prohibit the transmit/receive operation. When no communication is conducted, set CSIHnCTL0.CSIHnPWR to 0 to minimize power consumption of the CSIHn.
To start another transmission without stopping communication, perform steps 3 and later.

Section 14 Serial Communication Interface 3 (SCI3)

14.1 Overview

The serial communication interface 3 (SCI3: Serial Communication Interface 3) can handle two methods of serial communications: asynchronous and clock synchronous serial communications. Asynchronous serial data communications can be carried out with standard asynchronous communication LSIs such as Universal Asynchronous Receiver/Transmitter (UART) or Asynchronous Communication Interface Adapter (ACIA). Asynchronous mode is equipped with a serial communications function between multiple processors (multi-processor communications function).

14.1.1 Features of SCI3

Number of channels

This product contains the serial communication interface 3 for the number of channels shown below.

Table 14.1 Channels of SCI3

Serial Communication Interface 3	RH850/E1M-S
Number of channels	4
Name	SCI3n (n = 0 to 3)

Meaning of “n”

In this section, each channel of the serial communication interface 3 is identified by “n” (n = 0 to 3). For example, a serial mode register is written as SCI3nSMR.

14.1.2 Outline of Functions

- The serial data communication mode can be configured for asynchronous communications or clock synchronous communications.
- Full-duplex communications are available. The independent transmitter unit and receiver unit allow simultaneous transmission and reception. Both the transmitter and the receiver have a double-buffered structure, enabling continuous data transmission and reception.
- An arbitrary bit rate is selectable with the on-chip baud rate generator. An external clock is also selectable as a transmission/reception clock source.
- LSB-first or MSB-first transfer is selectable (except for asynchronous 7-bit data.)
- Interrupt sources: 4 types consisting of transmit-end, transmit-data-empty, receive-data-full, and receive-error. Transmit-data-empty and receive-data-full interrupt sources can activate the DMAC.
- Module stop mode can be configured.
- The bit rate modulation function can reduce errors averagely (even in a high bit rate) by correcting the output of the on-chip baud rate generator (except for the maximum speed in clock synchronous mode).
- The pin level of serial input data can be checked.
- On-chip 6-bit divider (PSC)

14.1.3 Serial Communication Modes

Asynchronous mode

- Data length: 7 bits or 8 bits selectable
- Stop bit length: 1 bit or 2 bits selectable
- Parity: Even parity, odd parity, or none electable
- Receive error detection: Parity error, overrun error, and framing error
- Break detection: A break can be detected by reading a register in the case of a framing error.

Clock synchronous mode

- Data length: 8 bits
- Receive error detection: Overrun error

14.1.4 Block Diagram

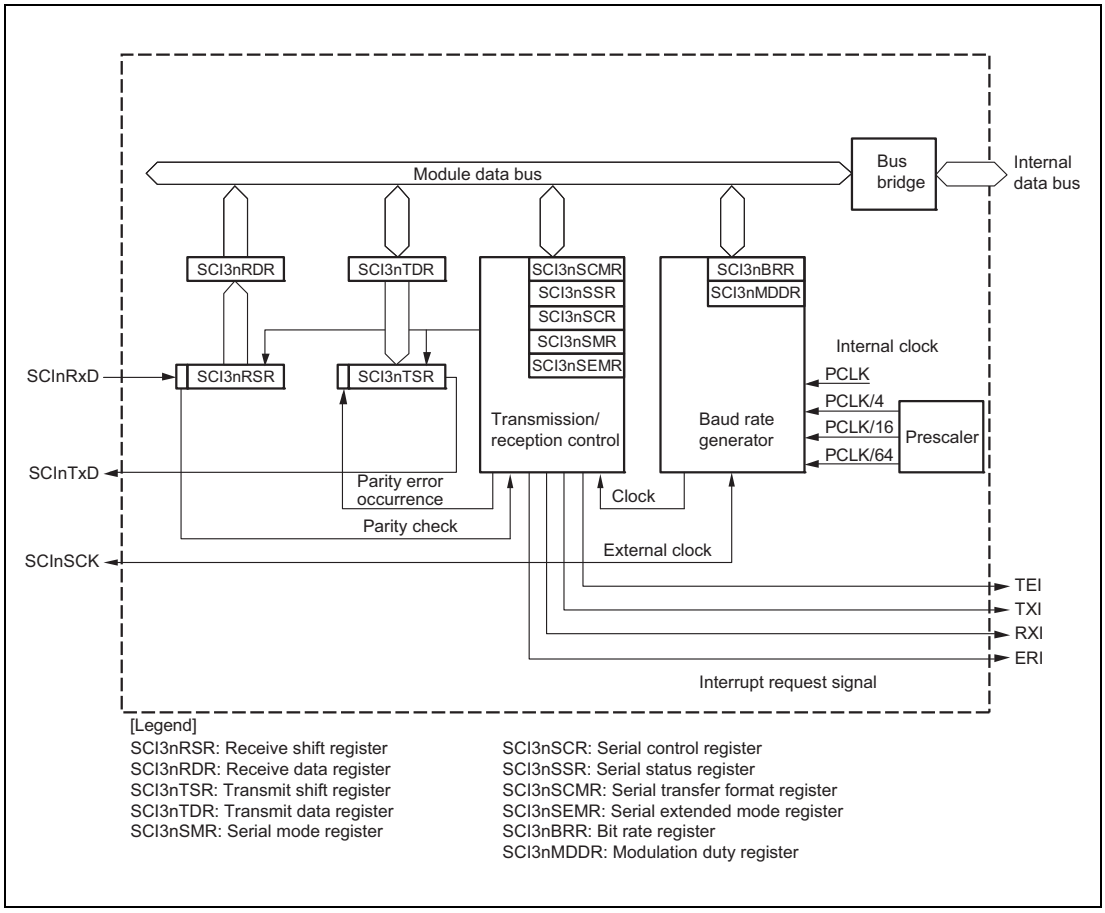


Figure 14.1 SCI3 Block Diagram

14.2 Input/Output Pins

The SCI3 has the input/output pins listed in **Table 14.2**.

Table 14.2 Pin Configuration

Name	Symbol	I/O	Function
Clock input/output pin	SCInSCK	Input	Serial clock input
		Output	Serial clock output
Receive data input pin	SCInRxD	Input	Receive data input
Transmit data output pin	SCInTxD	Output	Transmit data output

14.3 Register Descriptions

SCI3n register addresses are given as offsets from the base address <SCI3n_base>.

The base addresses <SCI3n_base> of each SCI3 are listed in **Table 14.3**.

Table 14.3 Register Base Address <SCI3n_base>

SCI3n Channel	SCI3n_base Address
SCI30	FFD9 0000_H
SCI31	FFD9 1000_H
SCI32	FFD9 2000_H
SCI33	FFD9 3000_H

The SCI3 has the following registers. Some registers have limitations in read/write by the CPU.

CAUTION

SCI3nBRR and SCI3nMDDR are allocated to the same address (relative address 4). The SCI3nMDDRS bit in SCI3nSEMR is used to switch these registers.

Table 14.4 Register Configuration

Register Name	Symbol Note:	Value after Reset	Address	Access Size
Receive shift register	SCI3nRSR	—	—	—
Serial mode register	SCI3nSMR	00_H	<SCI3n_base>+ 0000_H	8
Bit rate register / Modulation duty register	SCI3nBRR/ SCI3nMDDR	FF_H	<SCI3n_base>+ 0004_H	8
Serial control register	SCI3nSCR	00_H	<SCI3n_base>+ 0008_H	8
Transmit data register	SCI3nTDR	FF_H	<SCI3n_base>+ $000C_H$	8
Transmit shift register	SCI3nTSR	—	—	—
Serial status register	SCI3nSSR	84_H	<SCI3n_base>+ 0010_H	8
Receive data register	SCI3nRDR	00_H	<SCI3n_base>+ 0014_H	8
Serial transfer format register	SCI3nSCMR	$F2_H$	<SCI3n_base>+ 0018_H	8
Serial extended mode register	SCI3nSEMR	04_H	<SCI3n_base>+ $001C_H$	8

Note: n = 0 to 3

Relative addresses 4n + 1, 4n + 2, and 4n + 3 (n = 0 to 7) are reserved areas. These areas are always read as 0. Writing is invalid.

14.3.1 SCI3nRSR — Receive Shift Register

SCI3nRSR is a shift register which is used to receive serial data input from the SCInRxD pin and convert it to parallel data. When one frame of data has been received, it is automatically transferred to SCI3nRDR. SCI3nRSR cannot be directly accessed by the CPU.

14.3.2 SCI3nRDR — Receive Data Register

SCI3nRDR is an 8-bit register to store receive data. The initial value of SCI3nRDR is 00_H. When one frame of data has been received, it is transferred from SCI3nRSR to this register allowing SCI3nRSR to receive the next data. SCI3nRSR and SCI3nRDR function as a double-buffer in this way, allowing continuous receive operations. Be sure to check that the RDRF flag in SCI3nSSR is set to 1 before reading SCI3nRDR. SCI3nRDR cannot be written from the CPU.

When the data length is 7 bits, receive data is stored in bit 0 to 6 and bit 7 is fixed to 0 regardless of the SINV bit in SCI3nSCMR.

14.3.3 SCI3nTDR — Transmit Data Register

SCI3nTDR is an 8-bit register to store transmit data. The initial value of SCI3nTDR is FF_H. When SCI3nTSR empty is detected, the transmit data written to SCI3nTDR is transferred to SCI3nTSR and transmission starts. The double-buffered structure of SCI3nTDR and SCI3nTSR allows continuous serial transmission. If the next transmit data has been written to SCI3nTDR when one frame of data is sent, the transmit data is transferred to SCI3nTSR to continue transmission. SCI3nTDR can always be read and written by the CPU. Be sure to check that the TDRE flag in SCI3nSSR is set to 1 before writing transmit data to SCI3nTDR.

14.3.4 SCI3nTSR — Transmit Shift Register

SCI3nTSR is a shift register to transmit serial data. To perform serial data transmission, transmit data written to SCI3nTDR is automatically transferred SCI3nTSR, and is then sent to the SCInTxD pin. SCI3nTSR cannot be directly accessed by the CPU.

14.3.5 SCI3nSMR — Serial Mode Register

SCI3nSMR is a register used to select the transfer format and the clock source for the on-chip baud rate generator.

Access: This register can be read/written in 8-bit units.

Address: <SCI3n_base> + 0000_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	CM	CHR	PE	PM	STOP	MP	CKS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1

Note 1. Writable only when TE = RE = 0.

Table 14.5 SCI3nSMR Register Contents

Bit Position	Bit Name	Function
7	CM	Communications Mode 0: Asynchronous mode 1: Clock synchronous mode
6	CHR	Character Length (Valid only in asynchronous mode) 0: Selects 8 bits as the data length. 1: Selects 7 bits as the data length. LSB-first is fixed and the MSB (bit 7) in SCI3nTDR is not transmitted in transmission. In clock synchronous mode, the data length is fixed to 8 bits.
5	PE	Parity Enable (Valid only in asynchronous mode) When this bit is set to 1, a parity bit is added to transmit data and the parity bit is checked in reception. Regardless of the setting of this bit, no parity bit is added or checked in the multi-processor format.
4	PM	Parity Mode (Valid only when PE = 1 in asynchronous mode) 0: Selects even parity. 1: Selects odd parity. When even parity is set, parity bit is added so that the total number of 1 bits in the transmit/receive character plus the parity bit is even. Similarly, when odd parity is set, parity bit is added so that the total number of 1 bits in the transmit/receive character plus the parity bit is odd.
3	STOP	Stop Bit Length (Valid only in asynchronous mode) 0: 1 stop bit for transmission 1: 2 stop bits for transmission In reception, only the first stop bit is checked regardless of the setting of this bit. If the second stop bit is 0, it is treated as the start bit of the next transmission frame.
2	MP	Multi-Processor Mode (Valid only in asynchronous mode) When this bit is set to 1, the multi-processor communication function is enabled. In multi-processor mode, settings of the PE and PM bits are invalid.
1, 0	CKS[1:0]	Clock Select 1, 0 These bits select the clock source for the on-chip baud rate generator. 00: PCLK clock (n = 0) 01: PCLK/4 clock (n = 1) 10: PCLK/16 clock (n = 2) 11: PCLK/64 clock (n = 3) For the relation between the setting of these bits and the baud rate, see **Section 14.3.10, SCI3nBRR — Bit Rate Register**. The character n is the decimal notation of the value of n in **Section 14.3.10, SCI3nBRR — Bit Rate Register**.

14.3.6 SCI3nSCR — Serial Control Register

SCI3nSCR is a register used for the transmission/reception control, interrupt control, and transmission/reception clock source selection listed below. For interrupt requests, see **Section 14.8, Interrupt Sources**.

Access: This register can be read/written in 8-bit units.

Address: <SCI3n_base> + 0008_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	TIE	RIE	TE	RE	MPIE	TEIE	CKE[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W*1	R/W*1	R/W	R/W	R/W*2	R/W*2

Note 1. While the CM bit in SCI3nSMR is 1, a value of 1 can be written only when TE = 0 and RE = 0. After the TE or RE bit is set to 1, only 0 can be written to TE and RE. While the CM bit in SCI3nSMR is 0, writing is enabled at any timing.

Note 2. Writable only when TE = 0 and RE = 0. Also, writable at the same time when TE = 0 and RE = 0 are written.

Table 14.6 SCI3nSCR Register Contents (1/2)

Bit Position	Bit Name	Function
7	TIE	Transmit Interrupt Enable When this bit is set to 1, TXI interrupt request is enabled. TXI interrupt request can be cancelled by reading 1 from the TDRE flag and then clearing the flag to 0, or clearing the TIE bit.
6	RIE	Receive Interrupt Enable When this bit is set to 1, RXI and ERI interrupt requests are enabled. RXI and ERI interrupt requests can be cancelled by reading 1 from the RDRF, FER, PER, or ORER flag and then clearing the flag to 0, or clearing the RIE bit.
5	TE	Transmit Enable When this bit is set to 1, transmission is enabled. In this state, serial transmission is started by writing transmit data to SCI3nTDR and clearing the TDRE flag in SCI3nSSR to 0. Be sure to set SCI3nSMR before setting the TE bit to 1 to determine the transmission format. When this bit is set to 0 to disable transmission, the TDRE flag in SCI3nSSR is fixed to 1.
4	RE	Receive Enable When this bit is set to 1, reception is enabled. In this state, serial reception is started by detecting a start bit in asynchronous mode or the input of synchronous clock in clock synchronous mode. Be sure to set SCI3nSMR before setting the RE bit to 1 to determine the reception format. Even if reception is disabled by clearing this bit, the RDRF, FER, PER, and ORER flags are not affected and the previous value is retained.
3	MPIE	Multiprocessor Interrupt Enable (Valid only when MP in SCI3nSMR = 1 in asynchronous mode) When this bit is set to 1 and the data with the multi-processor bit set to 0 is received, the data is not read and setting the RDRF, FER, and ORER flags in SCI3nSSR to 1 is disabled. When the data with the multi-processor bit set to 1 is received, this bit is automatically cleared to 0, and normal reception is resumed. For details, see **Section 14.5, Multi-Processor Communication Function**. When the receive data includes MPB = 0 in SCI3nSSR, the receive data is not transferred from SCI3nRSR to SCI3nRDR, a receive error is not detected, and setting the RDRF, FER, and ORER flags in SCI3nSSR to 1 is disabled. When the receive data includes MPB = 1 in SCI3nSSR, the MPB bit in SCI3nSSR is set to 1, the MPIE bit is automatically cleared to 0, the RXI and ERI interrupt requests are enabled (if the RIE bit in SCI3nSCR is set to 1), and setting the FER and ORER flags to 1 is enabled.

Table 14.6 SCI3nSCR Register Contents (2/2)

Bit Position	Bit Name	Function
2	TEIE	Transmit End Interrupt Enable When this bit is set to 1, TEI interrupt request is enabled. TEI interrupt request can be cancelled by reading 1 from the TDRE flag and then clearing the flag to 0 to clear the TEND flag to 0, or clearing the TEIE bit.
1, 0	CKE[1:0]	Clock Enable 1, 0 These bits select the clock source and the SCInSCK pin function. For asynchronous mode 00: On-chip baud rate generator (The SCInSCK pin functions as an input/output port.) 01: On-chip baud rate generator (The clock with the same frequency as the bit rate is output from the SCInSCK pin.) 1X: Setting prohibited For clock synchronous mode 0X: Internal clock (The SCInSCK pin functions as a clock output pin.) 1X: External clock (The SCInSCK pin functions as a clock input pin.)

Note 1. X: Don’t care

NOTE

When writing to any bit other than the MPIE bit of this register, use a store instruction such that the value of the MPIE bit becomes 0.

Note that when using a bit-manipulation instruction for writing to this register, the read-modify-write operation may inadvertently set the MPIE bit to 1.

14.3.7 SCI3nSSR — Serial Status Register

SCI3nSSR consists of SCI3 status flags and transfer multi-processor bits.

The TDRE, RDRF, ORER, PER, and FER flags can be cleared only.

Access: This register can be read/written in 8-bit units.

Address: <SCI3n_base> + 0010_H

Value after reset: 84_H

Bit	7	6	5	4	3	2	1	0
	TDRE	RDRF	ORER	FER	PER	TEND	MPB	MPBT
Value after reset	1	0	0	0	0	1	0	0
R/W	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R	R	R/W

Note 1. Only 0 can be written to clear the flag.

Table 14.7 SCI3nSSR Register Contents (1/2)

Bit Position	Bit Name	Function
7	TDRE	Transmit Data Register Empty Indicates whether or not transmit data exists in SCI3nTDR. [Setting conditions] • When the TE bit in SCI3nSCR is 0 • When the data in SCI3nTDR has been transferred to SCI3nTSR so that new data can be written to SCI3nTDR [Clearing condition] • Writing 0 to TDRE after reading TDRE = 1 When transmit data is written to SCI3nTDR while TE = 1
6	RDRF	Receive Data Register Full Indicates whether or not receive data exists in SCI3nRDR [Setting condition] • When reception finishes successfully and the receive data is transferred from SCI3nRSR to SCI3nRDR [Clearing conditions] • Writing 0 to RDRF after reading RDRF = 1 • When data is read from SCI3nRDR Even when the RE bit in SCI3nSCR is cleared, the RDRF flag is not affected and the previous value is retained. • Note that completing the reception of the next data with the RDRF flag set to 1 will cause an overrun error, resulting in the loss of the receive data.
5	ORER	Overrun Error Indicates that an overrun error has occurred during reception and the reception abended. [Setting condition] • When the next data is received while RDRF = 1 In SCI3nRDR, receive data prior to an overrun error is retained, but data received after the overrun error occurrence is lost. When the ORER flag is set to 1, subsequent serial reception cannot be continued. In clock synchronous mode, serial transmission also cannot be continued. [Clearing condition] • Writing 0 to ORER after reading ORER = 1 Even when the RE bit in SCI3nSCR is cleared, the ORER flag is not affected and the previous value is retained.

Table 14.7 SCI3nSSR Register Contents (2/2)

Bit Position	Bit Name	Function
4	FER	Framing Error Indicates that a framing error has occurred during reception in asynchronous mode and the reception abended. [Setting condition] • When the stop bit is 0 In 2-stop-bit mode, only whether the first stop bit is 1 is checked but the second stop bit is not checked. Although the receive data when a framing error has occurred is transferred to SCI3nRDR, the RDRF flag is not set. In addition, while the FER flag is set to 1, the subsequent receive data is not transferred to SCI3nRDR. [Clearing condition] • Writing 0 to FER after reading FER = 1 Even when the RE bit in SCI3nSCR is cleared to 0, the FER flag is not affected and the previous value is retained.
3	PER	Parity Error Indicates that a parity error has occurred during reception in asynchronous mode and the reception abended. [Setting condition] • Although the receive data when a parity error has occurred is transferred to SCI3nRDR, the RDRF flag is not set. In addition, while the PER flag is set to 1, the subsequent receive data is not transferred to SCI3nRDR. [Clearing condition] • Writing 0 to PER after reading PER = 1 Even when the RE bit in SCI3nSCR is cleared to 0, the PER flag is not affected and the previous value is retained.
2	TEND	Transmit End [Setting conditions] • When the TE bit in SCI3nSCR is 0 • When the TDRE flag is 1 while the last bit of a transmit character is being transmitted [Clearing condition] • Writing 0 to the TDRE flag after reading TDRE = 1 • When transmit data is written to SCI3nTDR while TE = 1
1	MPB	Multi-Processor Bit Holds the value of the multi-processor bit in the received frame.
0	MPBT	Multi-Processor Bit Transfer Sets the value of the multi-processor bit to be added to the transmit frame.

14.3.8 SCI3nSCMR — Serial Transfer Format Register

SCI3nSCMR is a register to select transfer format for both asynchronous mode and clock synchronous mode.

Access: This register can be read/written in 8-bit units.

Address: <SCI3n_base> + 0018_H

Value after reset: $F2_H$

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	SDIR	SINV	—	—
Value after reset	1	1	1	1	0	0	1	0
R/W	R	R	R	R	R/W*1	R/W*1	R	R

Note 1. Writable only when TE = 0 and RE = 0.

Table 14.8 SCI3nSCMR Register Contents

Bit Position	Bit Name	Function
7 to 4	—	Reserved These bits are always read as 1. When writing, write the value after reset.
3	SDIR	Serial Data Transfer Direction (Valid in asynchronous mode and clock synchronous mode) This bit is used to select the direction of serial/parallel conversion. 0: Transfer with LSB-first 1: Transfer with MSB-first This bit is valid only when the transfer format is 8-bit data. For 7-bit data, LSB-first transfer is used.
2	SINV	Serial Data Invert (Valid in asynchronous mode and clock synchronous mode) This bit is used to invert the transfer data logic level. The SINV bit does not affect the logic level of the parity bit. To invert the parity bit, invert the PM bit in SCI3nSMR. 0: SCI3nTDR data is transmitted as it is, and receive data is stored in SCI3nRDR as it is. 1: SCI3nTDR data is inverted and transmitted, and receive data is inverted and stored in SCI3nRDR.
1	—	Reserved These bits are always read as 1. When writing, write the value after reset.
0	—	Reserved These bits are always read as 0. When writing, write the value after reset.

14.3.9 SCI3nSEMR — Serial Extended Mode Register

SCI3nSEMR is a register to select a 1-bit period.

Access: This register can be read/written in 8-bit units.

Address: <SCI3n_base> + $001C_H$

Value after reset: 04_H

Bit	7	6	5	4	3	2	1	0
	BRME	MDDRS	—	—	ABCS	RXDMON	—	—
Value after reset	0	0	0	0	0	1	0	0
R/W	R/W*1	R/W*1	R	R	R/W*1	R	R	R

Note 1. Writable only when TE = 0 and RE = 0.

Table 14.9 SCI3nSEMR Register Contents

Bit Position	Bit Name	Function
7	BRME	Bit Rate Modulation Enable When this bit is set to 1, the bit rate modulation function is enabled.
6	MDDRS	Modulation Duty Register Select This bit is used to select an accessible register. 0: SCI3nBRR is accessible. 1: SCI3nMDDR is accessible.
5, 4	—	Reserved These bits are always read as 0. When writing to these bits, write the value after reset.
3	ABCS	Asynchronous Reference Clock Select (Valid only in asynchronous mode) This bit is used to select the reference clock for 1-bit period. 0: Operates on the reference clock with a frequency of 16 times the transfer rate. 1: Operates on the reference clock with a frequency of 8 times the transfer rate (double-speed operation).
2	RXDMON	Serial Input Data Monitor This bit indicates the SCInRxD pin state. 0: SCInRxD pin is at the low level. 1: SCInRxD pin is at the high level.
1, 0	—	Reserved These bits are always read as 0. When writing to these bits, write the value after reset.

14.3.10 SCI3nBRR — Bit Rate Register

SCI3nBRR is an 8-bit register to adjust the bit rate. Since each SCI3 channel has an independent baud rate generator, different bit rates can be set for each channel. **Table 14.11** shows the relationship between the setting (N) in SCI3nBRR and the bit rate (B) in normal asynchronous mode and clock synchronous mode. The value after reset of SCI3nBRR is FF_H. SCI3nBRR is allocated at the same address as SCI3nMDDR and is selected when MDDRS in SCI3nSEMR is 0. This register is writable only when TE = 0 and RE = 0.

Access: This register can be read/written in 8-bit units.

Address: <SCI3n_base> + 0004_H

Value after reset: FF_H

Bit	7	6	5	4	3	2	1	0
	BRR							
Value after reset	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 14.10 SCI3nBRR Register Contents

Bit Position	Bit Name	Function
7 to 0	BRR	Baud rate generator setting (0 ≤ N ≤ 255)

Table 14.11 Relationship between Setting N in SCI3nBRR and Bit Rate B

Mode	ABCS Setting	Bit Rate	Mean Error
Asynchronous	0	$B = \frac{\phi \times 10^6}{64 \times 2^{2n-1} \times (N+1)}$	$Error\ (\%) = \{ \frac{\phi \times 10^6}{B \times 64 \times 2^{2n-1} \times (N+1)} - 1 \} \times 100$
	1	$B = \frac{\phi \times 10^6}{32 \times 2^{2n-1} \times (N+1)}$	$Error\ (\%) = \{ \frac{\phi \times 10^6}{B \times 32 \times 2^{2n-1} \times (N+1)} - 1 \} \times 100$
Clock synchronous	—	$B = \frac{\phi \times 10^6}{8 \times 2^{2n-1} \times (N+1)}$	

Note: B: Bit rate (bit/s)
N: SCI3nBRR setting for baud rate generator (0 ≤ N ≤ 255)
ϕ: Operating frequency PCLK (MHz)
n: Determined by the SCI3nSMR setting shown in the following table.

SCI3nSMR Setting		
CKS1	**CKS0**	**n**
0	0	0
0	1	1
1	0	2
1	1	3

Table 14.12 lists sample N settings of the SCI3nBRR register in asynchronous mode. **Table 14.13** lists the maximum settable bit rates.

Table 14.12 Examples of BRR Settings for Bit Rates (Asynchronous Mode)

	Operating Frequency ϕ = 40 (MHz)							
	SCI3nSEMR.ABCS = 0				SCI3nSEMR.ABCS = 1			
Bit Rate (bit/s)	n	N	Actual Bit Rate (bit/s)	Error (%)	n	N	Actual Bit Rate (bit/s)	Error (%)
110	3	177	109.73	–0.25	—	—	—	—
150	3	129	150.24	0.16	3	255	152.59	1.73
300	3	64	300.48	0.16	3	129	300.48	0.16
600	2	129	600.96	0.16	3	64	600.96	0.16
1200	2	64	1201.92	0.16	2	129	1201.92	0.16
2400	1	129	2403.85	0.16	2	64	2403.85	0.16
4800	0	255	4882.81	1.73	1	129	4807.69	0.16
9600	0	129	9615.38	0.16	1	64	9615.38	0.16
19200	0	64	19230.77	0.16	0	129	19230.77	0.16
31250	0	39	31250.00	0.00	0	79	31250.00	0.00
38400	0	32	37878.79	–1.36	0	64	38461.54	0.16

Table 14.13 Maximum Bit Rate (Asynchronous Mode)

	Setting			Maximum Bit Rate (bit/s)
ϕ (MHz)	ABCS Setting	n	N	
40	0	0	0	1250000
	1	0	0	2500000

Table 14.14 lists sample N settings of the SCI3nBRR register in clock synchronous mode. **Table 14.15** lists the supported maximum bit rates.

Table 14.14 Examples of Bit Rate Settings for Clock Synchronous Mode

	Operating Frequency ϕ = 40 (MHz)		
Bit Rate (bit/s)	**n**	**N**	**Actual Bit Rate (bit/s)**
1k	3	155	1001.60
2.5k	3	62	2480.16
5k	2	124	5000.00
10k	2	62	9920.63
25k	1	99	25000.00
50k	1	49	50000.00
100k	0	99	100000.00
250k	0	39	250000.00
500k	0	19	500000.00
1M	0	9	1000000.00
2M	0	4	2000000.00
2.5M	0	3	2500000.00
5M	0	1	5000000.00

Table 14.15 Maximum Bit Rate when Internal Clock is Output (Clock Synchronous Mode)

ϕ (MHz)	**n**	**N**	**Maximum Bit Rate (bit/s)**
40	0	1	5000000.00

14.3.11 SCI3nMDDR — Modulation Duty Register

SCI3nMDDR is a register to correct the bit rate adjusted by SCI3nBRR. The initial value of SCI3nMDDR is FF_H. When the BRME bit in SCI3nSEMR is set to 1, the bit rate generated by the on-chip baud rate generator is corrected to SCI3nMDDR/256 on average. **Table 14.17** shows the relationship between the SCI3nMDDR setting and the bit rate B. SCI3nMDDR is allocated at the same address as SCI3nBRR and is selected when MDDRS in SCI3nSEMR is 1. This register is writable only when TE = 0 and RE = 0. Bit 7 is fixed to 1.

Access: This register can be read/written in 8-bit units.

Address: <SCI3n_base> + 0004_H

Value after reset: FF_H

Bit	7	6	5	4	3	2	1	0
	MDDR							
Value after reset	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 14.16 SCI3nMDDR Register Contents

Bit Position	Bit Name	Function
7 to 0	MDDR	Baud rate generator setting (128 ≤ MDDR ≤ 255)

Table 14.17 Relationship between SCI3nMDDR Setting and Bit Rate B when Bit Rate Modulation Function Is Used

Mode	ABCS Setting	Bit Rate	Mean Error
Asynchronous	0	$B = \frac{\phi \times 10^6}{64 \times 2^{2n-1} \times (256/MDDR) \times (N+1)}$	$Error\ (\%) = \{\frac{\phi \times 10^6}{B \times 64 \times 2^{2n-1} \times (256/MDDR) \times (N+1)} - 1\} \times 100$
	1	$B = \frac{\phi \times 10^6}{32 \times 2^{2n-1} \times (256/MDDR) \times (N+1)}$	$Error\ (\%) = \{\frac{\phi \times 10^6}{B \times 32 \times 2^{2n-1} \times (256/MDDR) \times (N+1)} - 1\} \times 100$
Clock synchronous	—	$B = \frac{\phi \times 10^6}{8 \times 2^{2n-1} \times (256/MDDR) \times (N+1)}$	

Note: B: Bit rate (bit/s)
N: SCI3nBRR setting of baud rate generator (0 ≤ N ≤ 255)
ϕ: Operating frequency PCLK (MHz)
n: See **Table 14.11, Relationship between Setting N in SCI3nBRR and Bit Rate B**.
SCI3nMDDR:SCI3nMDDR setting (128 ≤ SCI3nMDDR ≤ 255)

14.4 Operation in Asynchronous Mode

Figure 14.2 shows the general format for asynchronous serial communications. One frame consists of a start bit (low level), transmit/receive data, a parity bit, and stop bits (high level). In asynchronous serial communications, the communication line is usually held in the mark state (high level). The SCI3 monitors the communication line, and when the SCI3 detects the space state (low level), it recognizes a start bit and starts serial communications. Inside the SCI3, the transmitter and the receiver are independent, enabling full-duplex communications. Both the transmitter and the receiver have a double-buffered structure so that data can be read or written during transmission or reception, enabling continuous data transmission and reception.

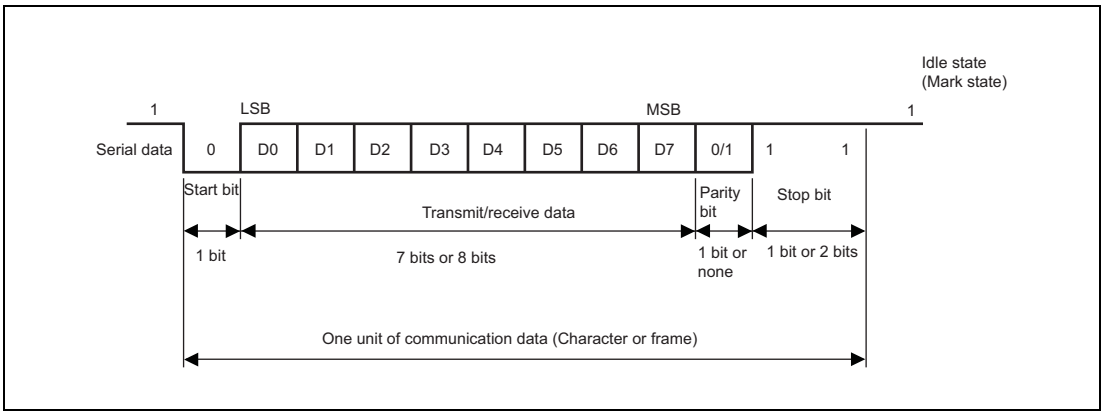


Figure 14.2 Data Format in Asynchronous Serial Communications (Example with 8-Bit Data, Parity, Two Stop Bits)

14.4.1 Transmission/Receive Format

Table 14.18 lists settable transmission/reception formats in asynchronous mode. Any of 12 transmission/reception formats can be selected according to the SCI3nSMR setting. For details of the multi-processor bit, see **Section 14.5, Multi-Processor Communication Function**.

Table 14.18 Serial Transmission/Receive Formats (Asynchronous Mode)

SMR Setting				Serial Transmission/Reception Format and Frame Length
CHR	PE	MP	STOP	1 2 3 4 5 6 7 8 9 10 11 12
0	0	0	0	S 8-bit data STOP
0	0	0	1	S 8-bit data STOP STOP
0	1	0	0	S 8-bit data P STOP
0	1	0	1	S 8-bit data P STOP STOP
1	0	0	0	S 7-bit data STOP
1	0	0	1	S 7-bit data STOP STOP
1	1	0	0	S 7-bit data P STOP
1	1	0	1	S 7-bit data P STOP STOP
0	—	1	0	S 8-bit data MPB STOP
0	—	1	1	S 8-bit data MPB STOP STOP
1	—	1	0	S 7-bit data MPB STOP
1	—	1	1	S 7-bit data MPB STOP STOP

Note: S: Start bit
STOP: Stop bit
P: Parity bit
MPB: Multi-processor bit

14.4.2 Receive Data Sampling Timing and Receive Margin

In asynchronous mode, the SCI3 operates on a reference clock with a frequency of 16 times (8 times for the double-speed mode) the bit rate. In reception, the SCI3 samples the falling edge of the beginning of the start bit (low level) using the reference clock and performs internal synchronization. As shown in **Figure 14.3**, data is latched at the middle of each bit by sampling receive data at the rising edge of the eighth pulse (fourth pulse for the double-speed mode) of the reference clock.

Thus the reception margin in asynchronous mode is determined by formula (1) below.

$$M = \left| (0.5 - \frac{1}{2N}) - (L - 0.5)F - \frac{|D - 0.5|}{N}(1 + F) \right| \times 100\ [\%] \quad \cdots \text{formula (1)}$$

M: Receive margin
N: Ratio of bit rate to clock (N = 16 when ABCS in SCI3nSEMR = 0, N = 8 when ABCS = 1)
D: Duty cycle of clock (D = 0.5 to 1.0)
L: Frame length (L = 9 to 12)
F: Absolute value of clock frequency deviation

Assuming that F (absolute value of clock frequency deviation) = 0, D (duty cycle of clock) = 0.5, and N = 16 in formula (1), the reception margin is obtained by the formula below.

$M = \{0.5 - 1/(2 \times 16)\} \times 100\ [\%] = 46.875\%$

However, this is only the calculated value, and a margin of 20% to 30% should be allowed in system design.

When the bit rate modulation function is used, the reference clock frequency is corrected on average.

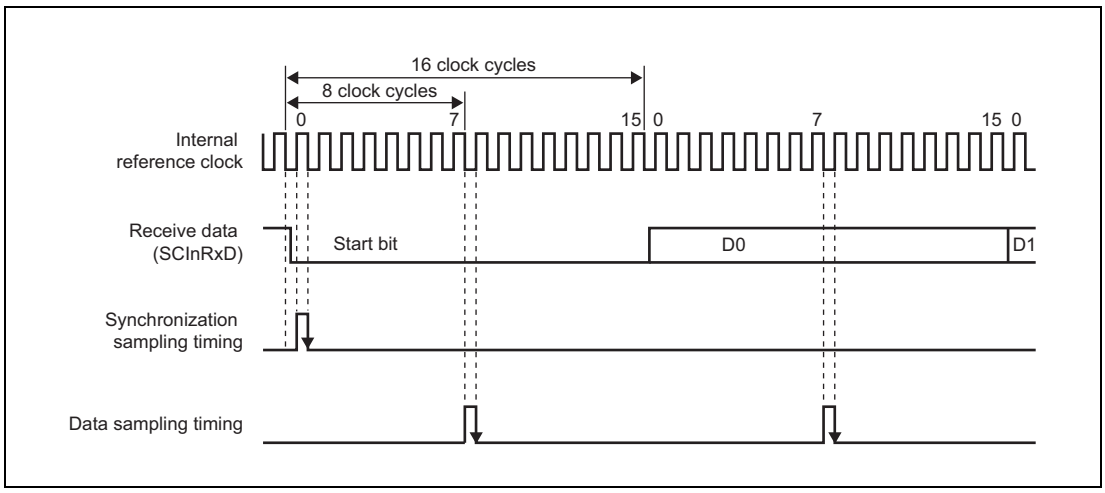


Figure 14.3 Receive Data Sampling Timing in Asynchronous Mode

14.4.3 Clock

An internal clock generated by the on-chip baud-rate generator can be set as the SCI3's transmission/ reception clock according to the settings of the CM bit in SCI3nSMR and the CKE1 and CKE0 bits in SCI3nSCR. When the SCI3 is operated on an internal clock, the clock can be output from the SCInSCK pin.

For details of clock synchronous mode, see **Section 14.6, Operation in Clock Synchronous Mode**.

In asynchronous mode, the frequency of the clock output is equal to the bit rate and the phase is such that the rising edge of the clock comes at the center of the transmit data, as shown in **Figure 14.4**.

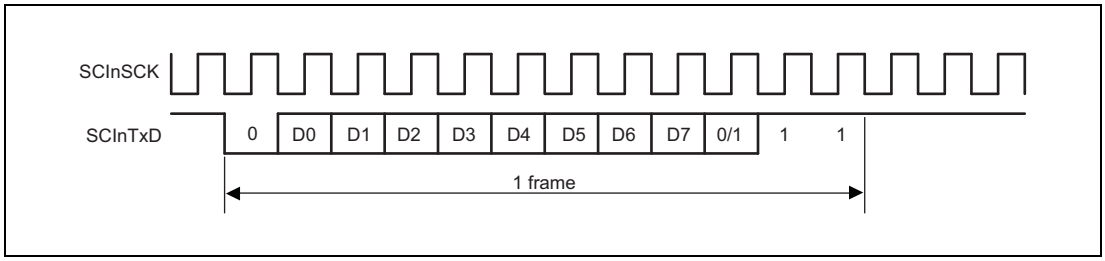


Figure 14.4 Phase Relationship between Output Clock and Transmit Data (Asynchronous Mode)

14.4.4 Double-Speed Operation

In addition to the operation described in **Section 14.4.3, Clock**, double-speed operation is enabled by the setting of the ABCS bit in SCI3nSEMR.

In double-speed operation, the same operation on a clock with a frequency of 16 times the bit rate in normal operation can be conducted on a clock with a frequency of 8 times the bit rate, meaning that the SCI3 operates on a double transfer rate using the same reference clock.

Double-speed operation can be configured with an internal clock generated by the on-chip baud-rate generator.

14.4.5 SCI3 Initialization (Asynchronous Mode)

Before transmitting and receiving data, clear the TE and RE bits in SCI3nSCR and then initialize the SCI3 according to the sample flowchart in **Figure 14.5**. Before changing the operating mode or transfer format, be sure to clear the TE and RE bits to 0. Note that clearing the TE bit to 0 sets the TDRE flag to 1, but clearing the RE bit to 0 initializes neither the RDRF, PER, FER, and ORER flags nor SCI3nRDR.

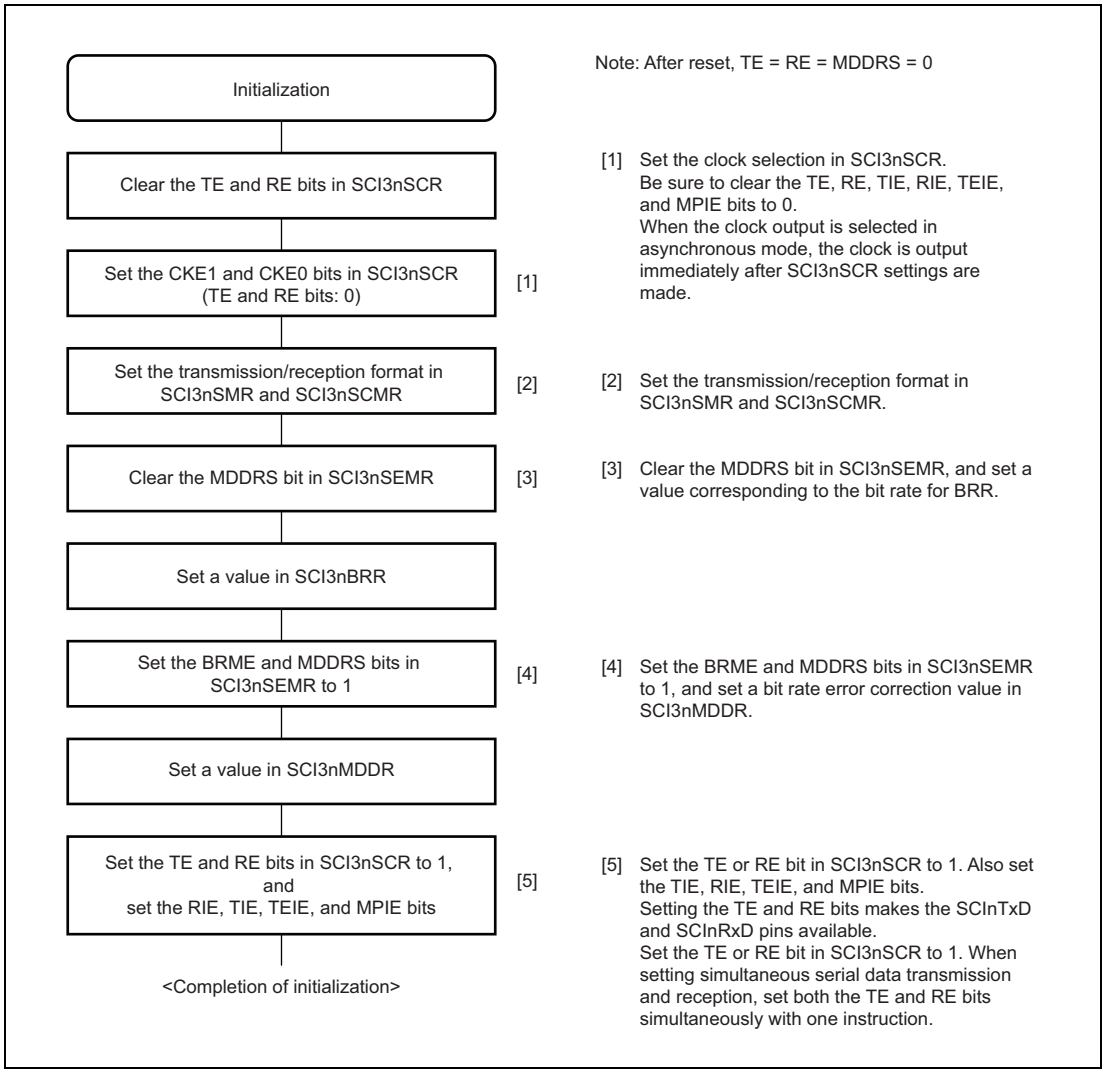


Figure 14.5 Sample Flowchart for SCI3 Initialization

14.4.6 Serial Data Transmission (Asynchronous Mode)

Figure 14.6 shows an example of operation for data transmission in asynchronous mode. In data transmission, the SCI3 operates as described below.

1. When transmit data is written to SCI3nTDR, the TDRE flag is automatically cleared to 0. The SCI3 monitors the TDRE flag in SCI3nSSR. When the flag is cleared, the SCI3 recognizes that data has been written to SCI3nTDR and transfers data from SCI3nTDR to SCI3nTSR. When writing transmit data to SCI3nTDR at a trigger of TXI interrupt request, set the TIE bit to 1 and then set the TE bit to 1 or set both TIE and TE bits simultaneously with one instruction to generate a TXI interrupt request for starting data transfer.
2. Transmission starts after data is transferred from SCI3nTDR to SCI3nTSR and the TDRE flag is set to 1. If the TIE bit in SCI3nSCR is set to 1 at this time, a TXI interrupt request is generated. Continuous transmission is enabled by writing the next transmit data to SCI3nTDR in this TXI interrupt processing routine before transmission of the previously transferred data is completed. When using a TEI interrupt request, write the last transmit data to SCI3nTDR and then clear the TIE bit to 0 and set the TEIE bit to 1.
3. Data is sent from the SCInTxD pin in the following order: start bit, transmit data, parity bit or multi-processor bit (may be omitted depending on the format), and stop bit.
4. The TDRE flag is checked when the stop bit is output.
5. When the TDRE flag is 0, the next transmit data is transferred from SCI3nTDR to SCI3nTSR. After the stop bit has been sent, transmission of the next frame starts.
6. When the TDRE flag is 1, the TEND flag in SCI3nSSR is set to 1, the stop bit is sent, and then 1 is output to enter the mark state. If the TEIE bit in SCI3nSCR is set to 1 at this time, a TEI interrupt request is generated.

Figure 14.7 shows a sample flowchart for data transmission. **Figure 14.8** shows a sample flowchart for stopping the SCI3 after data transmission.

Supplementary note on operation when data transmission in asynchronous mode is enabled

When the TE bit is set to 1, the high level (preamble) is output for a frame. When transmit data is written to SCI3nTDR while the preamble is output, the data is transferred from SCI3nTDR to SCI3nTSR following the completion of the preamble output.

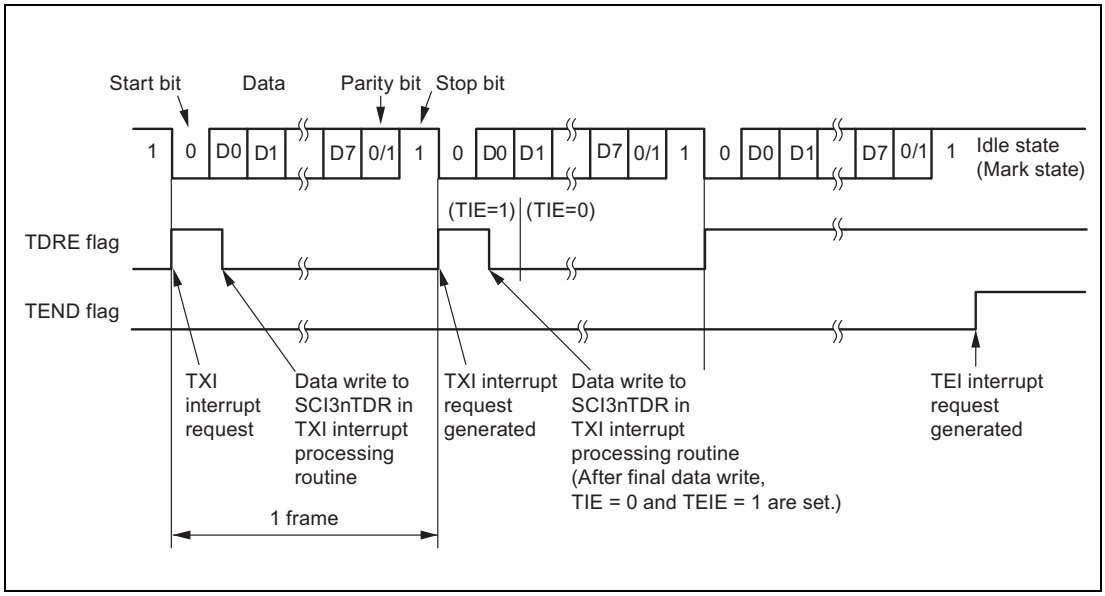


Figure 14.6 Example of Operation for Transmission in Asynchronous Mode (Example with 8-Bit Data, Parity, One Stop Bit)

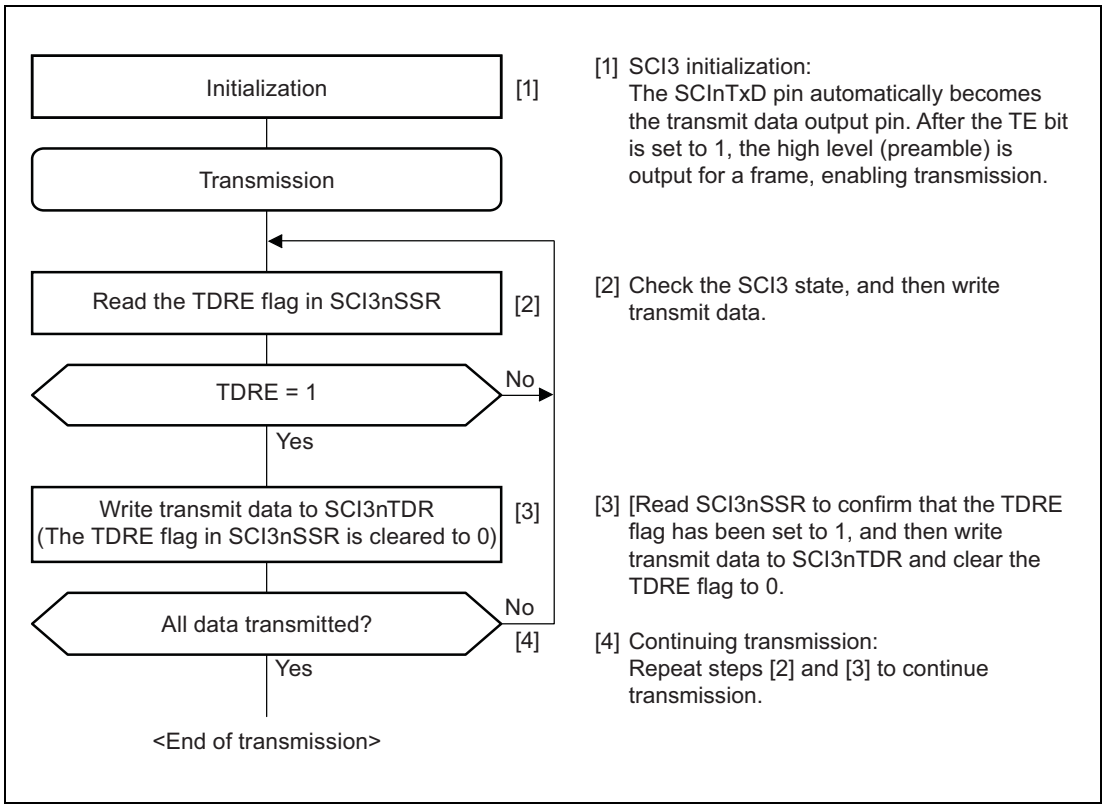
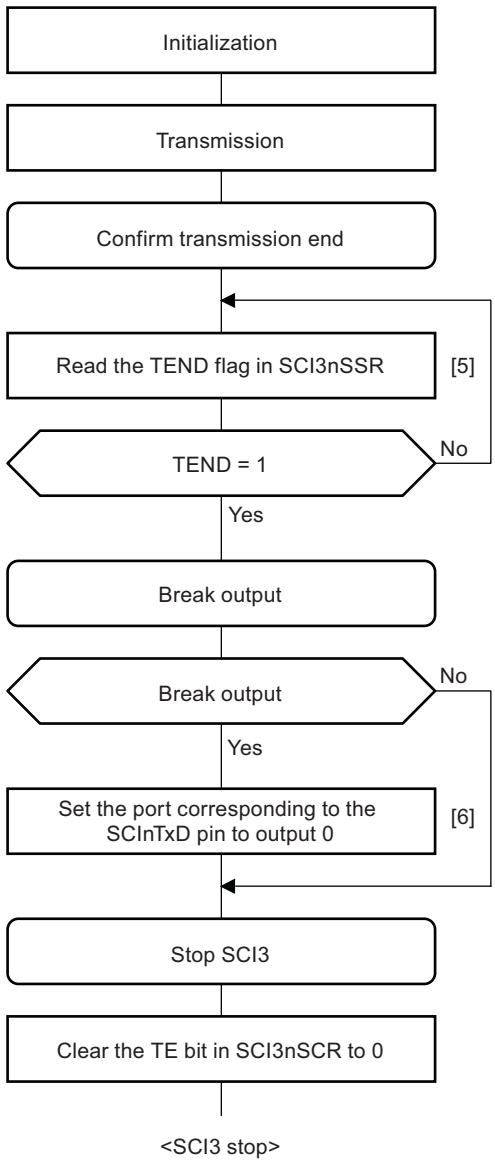


Figure 14.7 Example of Serial Transmission Flowchart



[5] Confirmation of transmission end:
Read the TEND flag in SCI3nSSR to confirm that no data is being transmitted.

[6] To output a break at the end of transmission, set the port corresponding to the SCInTxD pin to port output mode to output the low level (clear the P, PM, and PMC bits to 0), and then clear the TE and RE bits in SCI3nSCR to 0.

Note 1. Clearing the TE bit in SCI3nSCR initializes the SCI3 transmitter and places the SCInTxD pin in the high-impedance state. To retain the serial communications line, switch the SCInTxD pin to port output mode to output the required level before clearing the TE bit to 0.

Figure 14.8 Example Flowchart for Stopping the SCI3 after Serial Transmission

14.4.7 Serial Data Receive (Asynchronous Mode)

Figure 14.9 shows an example of the operation for data reception in asynchronous mode. In data reception, the SCI3 operates as described below.

1. When the SCI3 monitors the communications line and detects a start bit, it performs internal synchronization, stores receive data in SCI3nRSR, and checks the parity bit and the stop bit.
2. When an overrun error occurs (the next data has been received with the RDRF flag in SCI3nSSR set to 1), the ORER flag in SCI3nSSR is set to 1. If the RIE bit in SCI3nSCR is set to 1 at this time, an ERI interrupt request is generated. Receive data is not transferred to SCI3nRDR. The RDRF flag retains the state of being set to 1.
3. When a parity error is detected, the PER flag in SCI3nSSR is set to 1 and receive data is transferred to SCI3nRDR. If the RIE bit in SCI3nSCR is set to 1 at this time, an ERI interrupt request is generated.
4. When a framing error (when the stop bit is 0) is detected, the FER flag in SCI3nSSR is set to 1 and receive data is transferred to SCI3nRDR. If the RIE bit in SCI3nSCR is set to 1 at this time, an ERI interrupt request is generated.
5. When reception finishes successfully, the RDRF flag in SCI3nSSR is set to 1 and receive data is transferred to SCI3nRDR. If the RIE bit in SCI3nSCR is set to 1 at this time, an RXI interrupt request is generated. Continuous reception is enabled by reading the receive data transferred to SCI3nRDR in this RXI interrupt processing routine before reception of the next receive data is completed. Reading SCI3nRDR automatically clears the RDRF flag to 0.

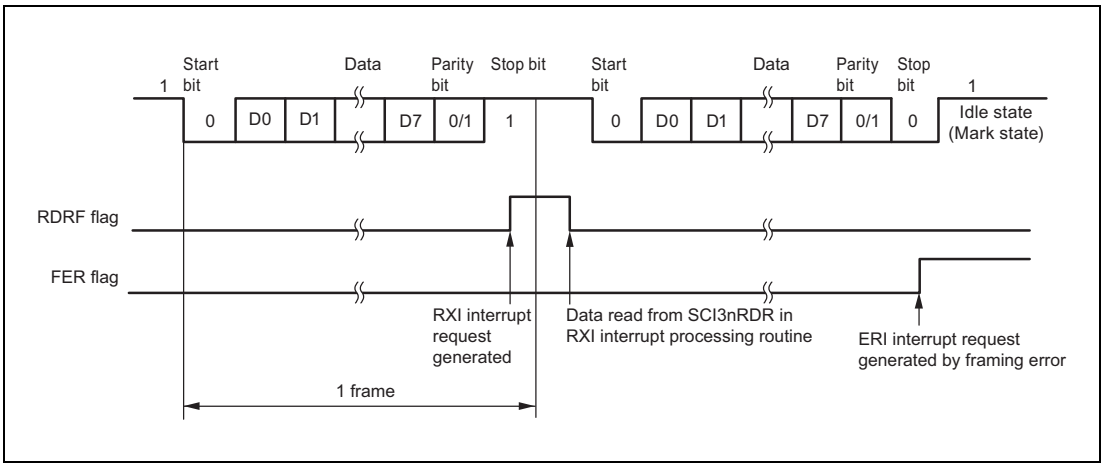


Figure 14.9 Example of Operation for Receive in Asynchronous Mode (Example with 8-Bit Data, Parity, One Stop Bit)

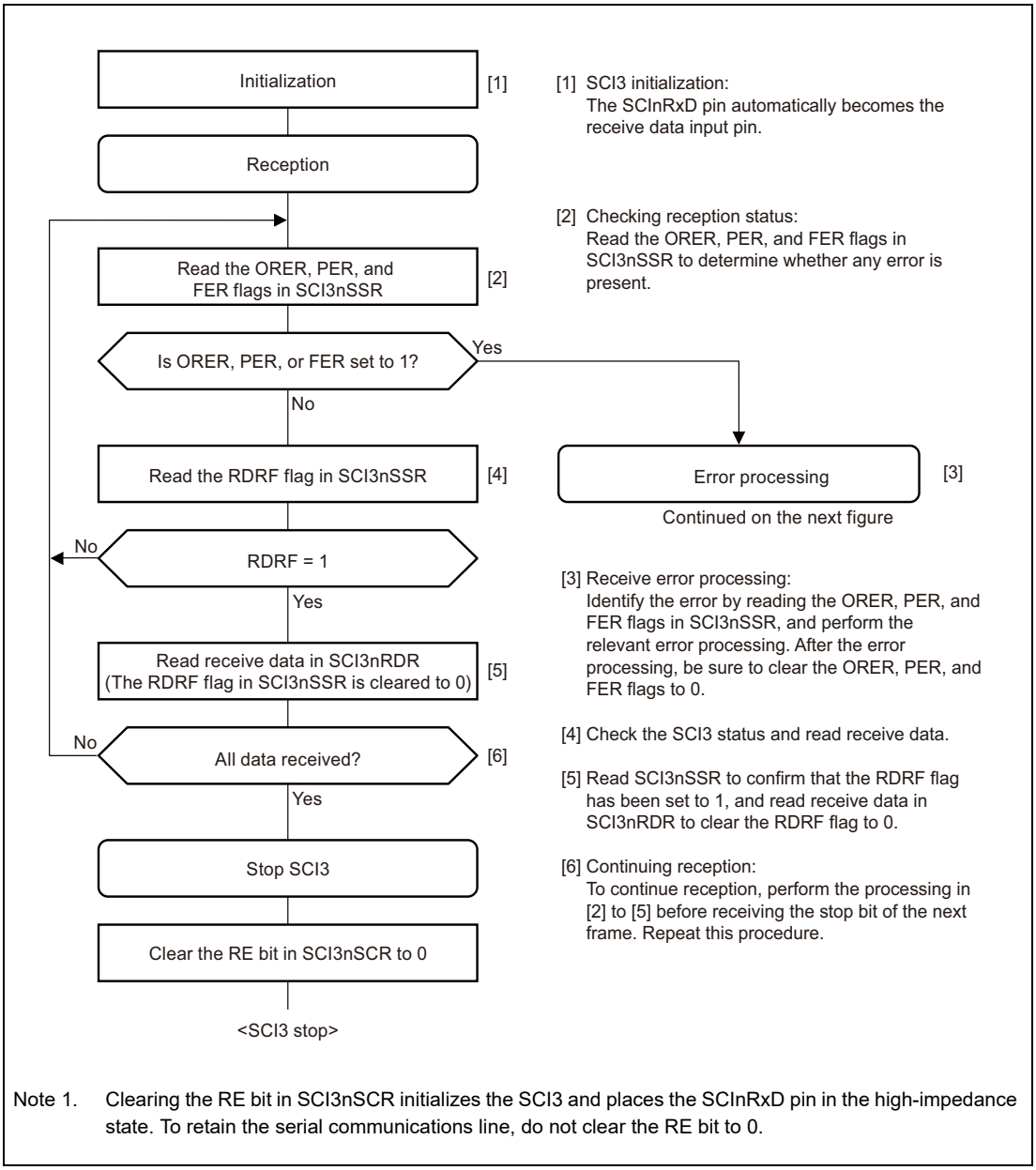
Table 14.19 lists the states of the SCI3nSSR status flags and receive data handling when a receive error is detected. When a receive error is detected, the RDRF flag retains the status before receiving the data. Subsequent data reception is disabled while a receive error flag is set to 1. Accordingly, clear the ORER, FER, PER, and RDRF flags before continuing data reception. **Figure 14.10** shows a sample flowchart for data reception.

Table 14.19 SCI3nSSR Status Flags and Receive Data Handling

SCI3nSSR Status Flags					
RDRF*1	ORER	FER	PER	Receive Data	Receive Status
1	0	0	0	Transferred to SCI3nRDR	Successful reception
0	0	1	0	Transferred to SCI3nRDR	Framing error
0	0	0	1	Transferred to SCI3nRDR	Parity error
0	0	1	1	Transferred to SCI3nRDR	Framing error + parity error
1*	1	0	0	Lost	Overrun error
1*	1	1	0	Lost	Overrun error + framing error
1*	1	0	1	Lost	Overrun error + parity error
1*	1	1	1	Lost	Overrun error + framing error + parity error

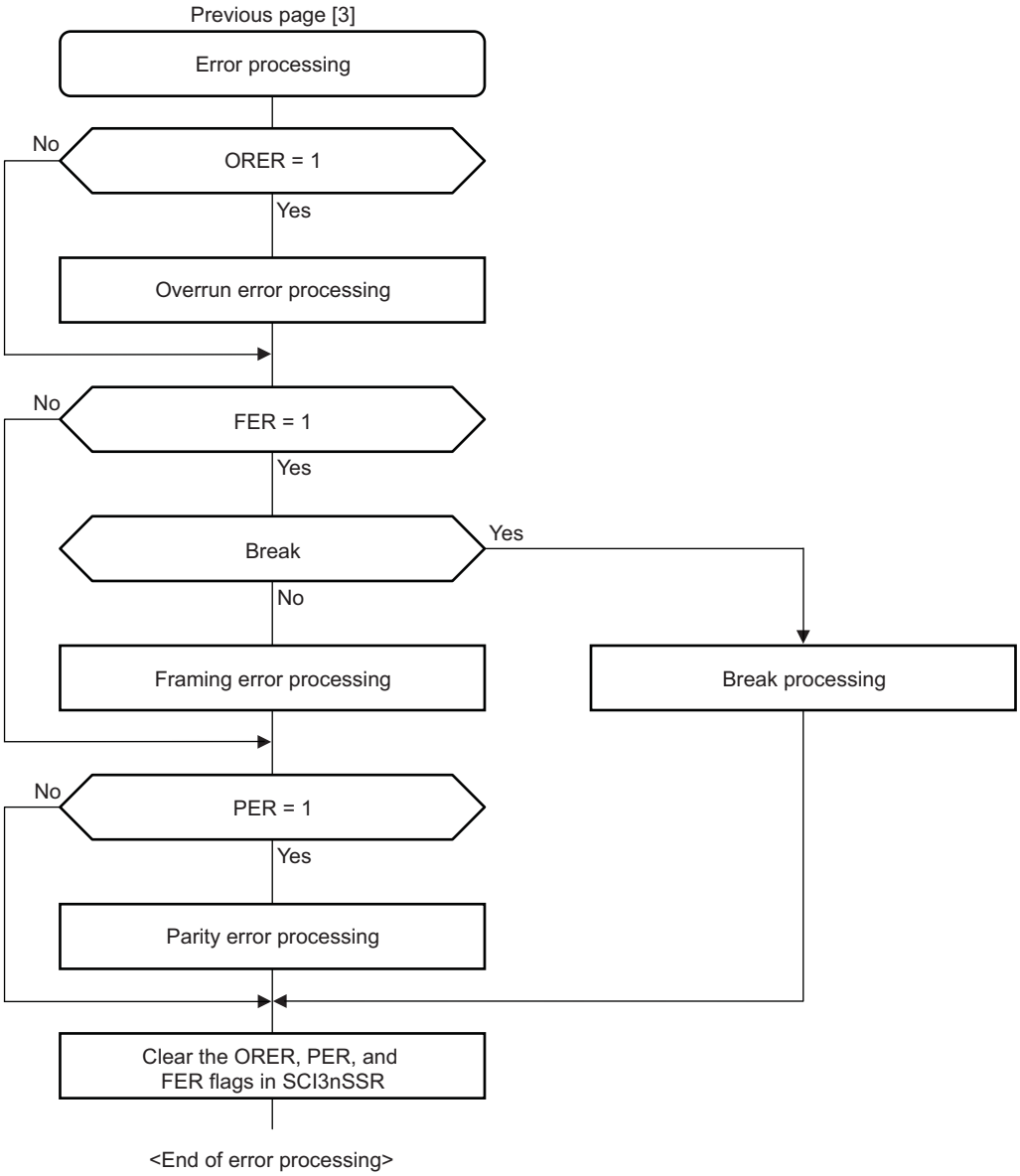
Note 1. In the case of an overrun error, the RDRF flag retains the state before the data reception.

Note: A sign “+” indicates that two or more receive states occur simultaneously in a single reception operation.



Note 1. Clearing the RE bit in SCI3nSCR initializes the SCI3 and places the SCInRxD pin in the high-impedance state. To retain the serial communications line, do not clear the RE bit to 0.

Figure 14.10 Example of Serial Receive Flowchart (1)



Note 1. In the case of a framing error, a break can be detected by reading the value of the RXDMON bit in SCI3nSEMR.

Note 2. After error processing, be sure to clear the ORER, PER, and FER flags to 0. Receive cannot be resumed if either of these flags is set to 1.

Figure 14.11 Example of Serial Receive Flowchart (2)

14.5 Multi-Processor Communication Function

14.5.1 Overview and Sample Connection

Using the multi-processor communication function allows data transmission and reception by sharing a communication line between multiple processors by using asynchronous serial communication in which the multi-processor bit is added. In multi-processor communication, a unique ID code is allocated to each receiving station. Serial communication cycles consist of an ID transmission cycle to specify the receiving station and a data transmission cycle to transmit data to the specified receiving station. The multi-processor bit is used to distinguish the ID transmission cycle from the data transmission cycle. When the multi-processor bit is set to 1, it indicates the ID transmission cycle. When the multi-processor bit is set to 0, it indicates the data transmission cycle. **Figure 14.12** shows an example of communication between processors by using the multi-processor format. First, a transmitting station sends communication data in which the multi-processor bit (= 1) is added to the ID code of the receiving station. Next, the transmitting station sends communication data in which the multi-processor bit (= 0) is added to the transmit data. Upon receiving the communication data in which the multi-processor bit is set to 1, the receiving station compares the received ID with the ID of the receiving station itself. When these IDs match, the receiving station receives communication data that is subsequently transmitted. If these IDs do not match, the receiving station skips communication data until it receives communication data in which the multi-processor bit is set to 1.

To support this function, the SCI3 provides the MPIE bit in SCI3nSCR. When the MPIE bit is set to 1, transfer of receive data from SCI3nRSR to SCI3nRDR, detection of a receive error, and setting the RDRF, FER, and ORER flags in SCI3nSSR are disabled until data in which the multi-processor bit is set to 1 is received. Upon receiving a character in which the multi-processor bit is set to 1, the MPB bit in SCI3nSSR is set to 1 and the MPIE bit is automatically cleared to 0, thus returning to a normal reception operation. When the RIE bit in SCI3nSCR is set to 1 at this time, an RXI interrupt request is generated. While the MPIE bit is cleared to 0, reception operation is conducted regardless of the multi-processor bit value. The multi-processor bit is stored in the MPB bit in SCI3nSSR.

When the multi-processor format is specified, specification of the parity bit is disabled. Apart from this, there is no difference from the operation in the normal asynchronous mode. A clock that is used for multi-processor communications is also the same as the clock used in the normal asynchronous mode.

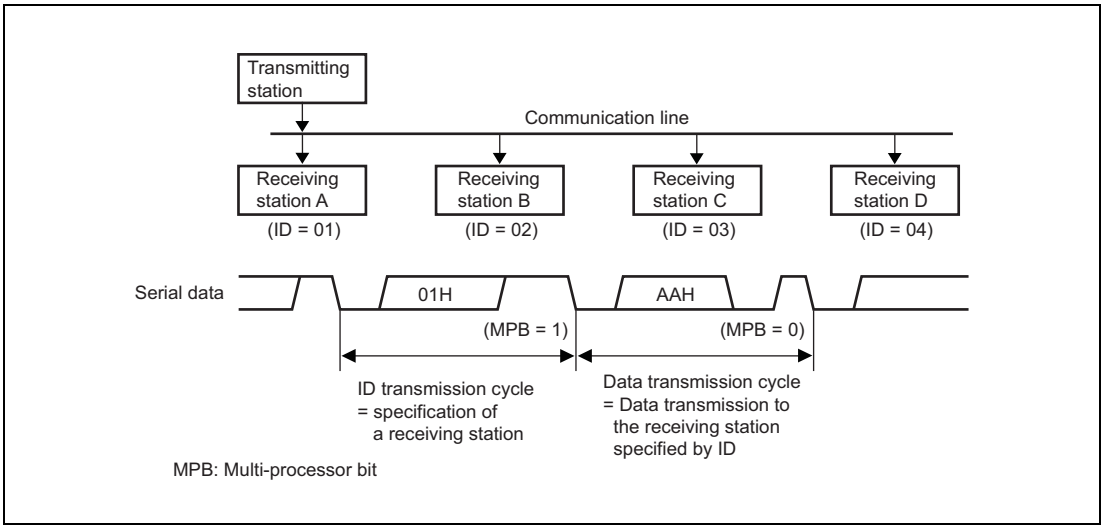


Figure 14.12 Example of Communication Using the Multi-Processor Format (Example of Transmission of Data AA_H to Receiving Station A)

14.5.2 Multi-Processor Serial Data Transmission

Figure 14.13 shows a sample flowchart of multi-processor data processing. In the ID transmission cycle, send the ID with the MPBT bit in SCI3nSSR set to 1. In the data transmission cycle, send data with the MPBT bit in SCI3nSSR cleared to 0. Other operations are the same as operations in asynchronous mode.

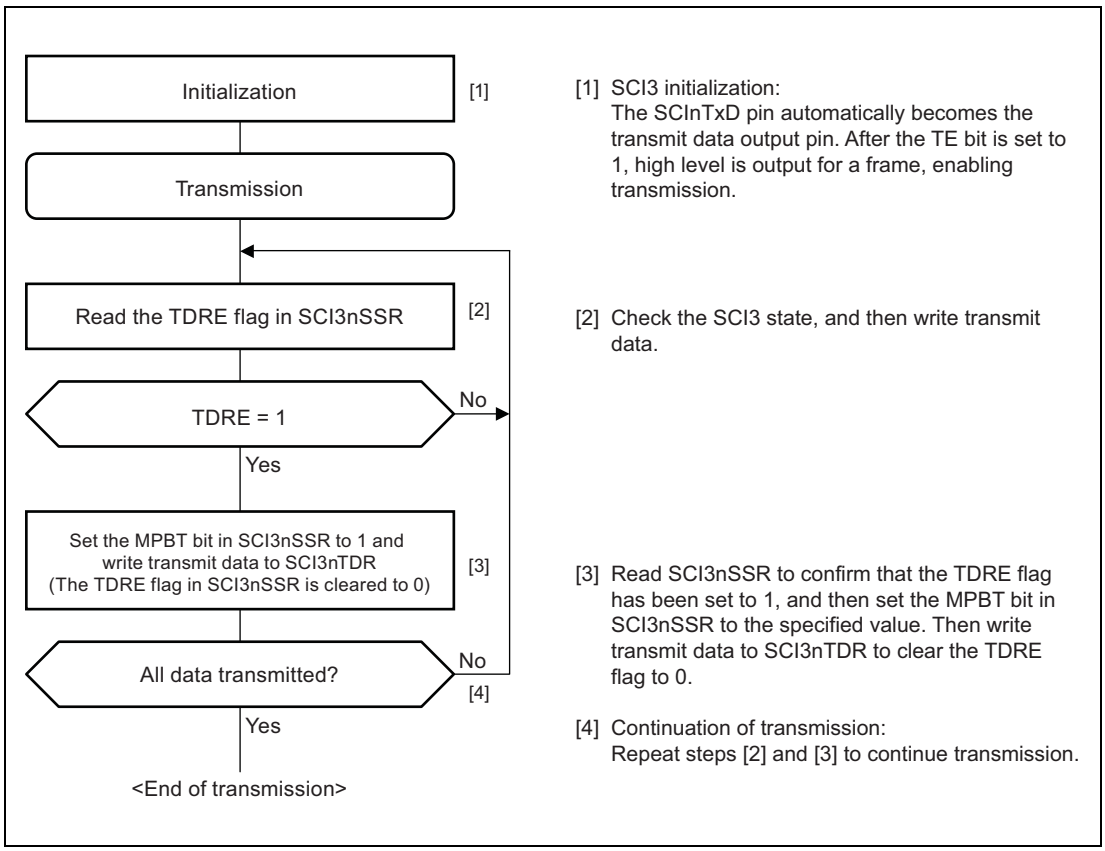


Figure 14.13 Example of Multi-Processor Serial Transmission Flowchart

14.5.3 Multi-Processor Serial Data Receive

Figure 14.15 shows sample flowcharts of multi-processor data reception. When the MPIE bit in SCI3nSCR is set to 1, reading the communication data is skipped until communication data in which the multi-processor bit is set to 1 is received. When communication data in which the multi-processor bit is set to 1 is received, the receive data is transferred to SCI3nRDR. At this time, an RXI interrupt request is generated. Other operations are the same as operations in asynchronous mode. **Figure 14.14** shows an example of operation for reception.

CAUTION

Do not write data to SCI3nSCR when communication data in which the multi-processor bit is set to 1 is received. The MPIE bit may not become the desired state.

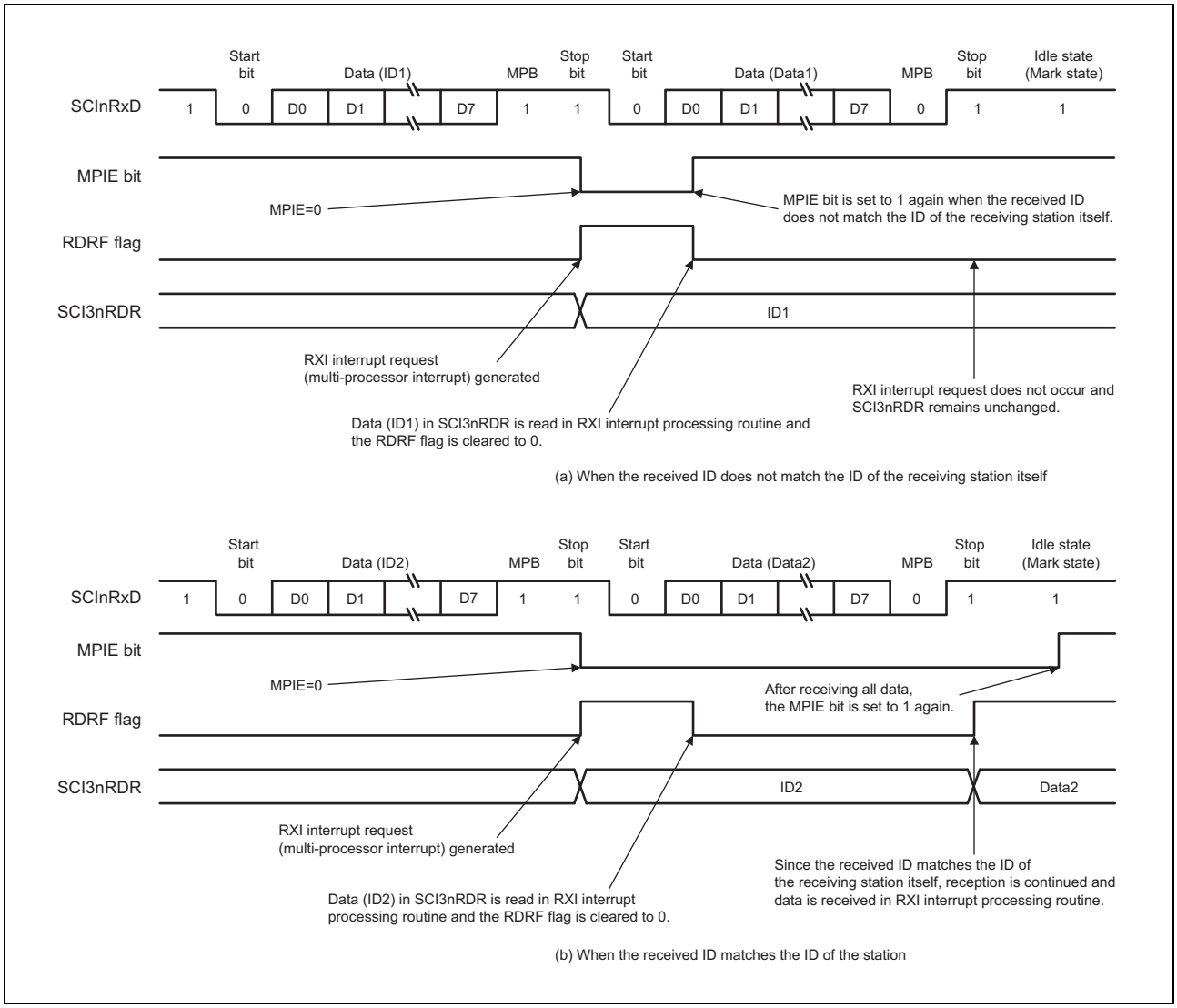


Figure 14.14 Example of SCI3 Receive (8-Bit Data, Multi-Processor Bit, One Stop Bit)

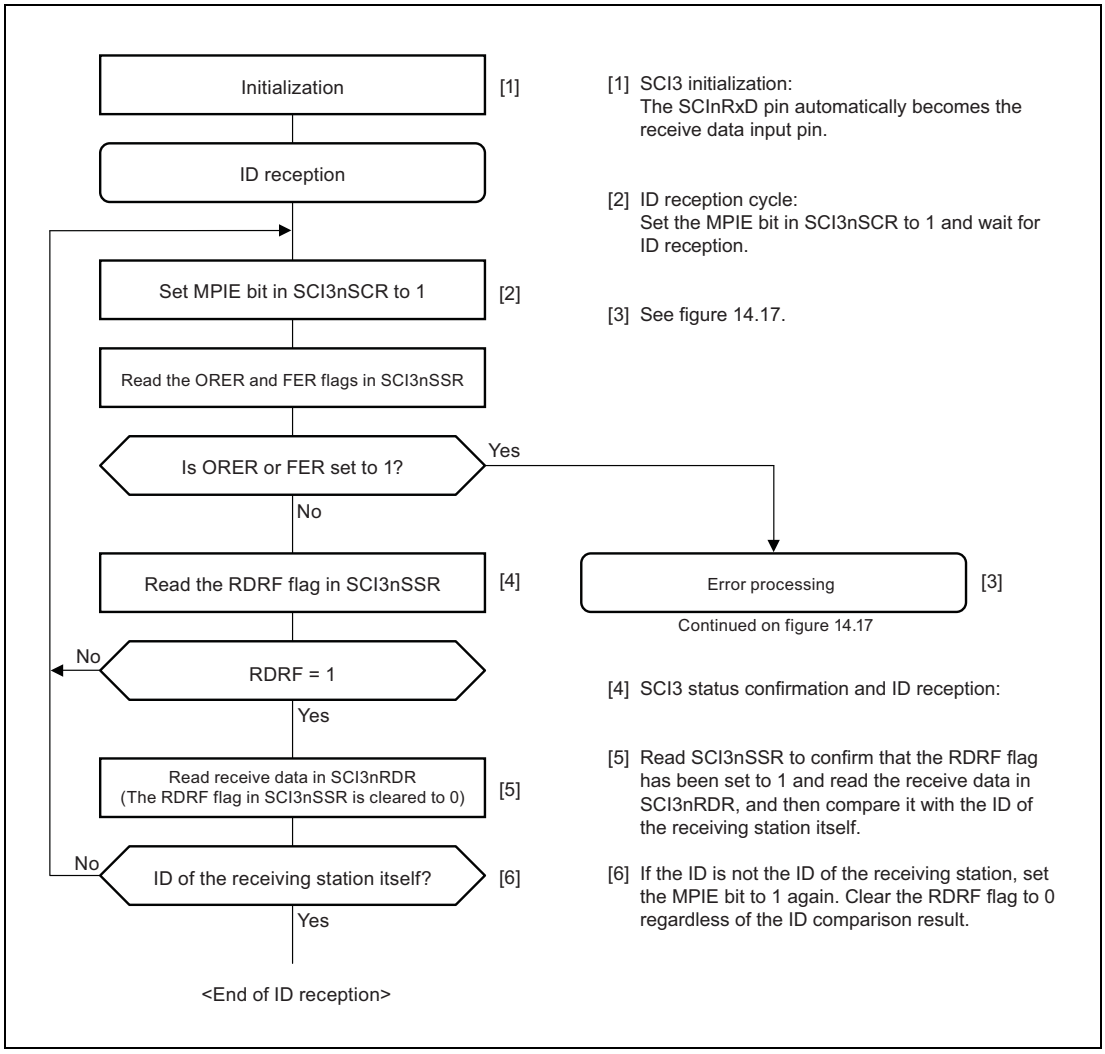
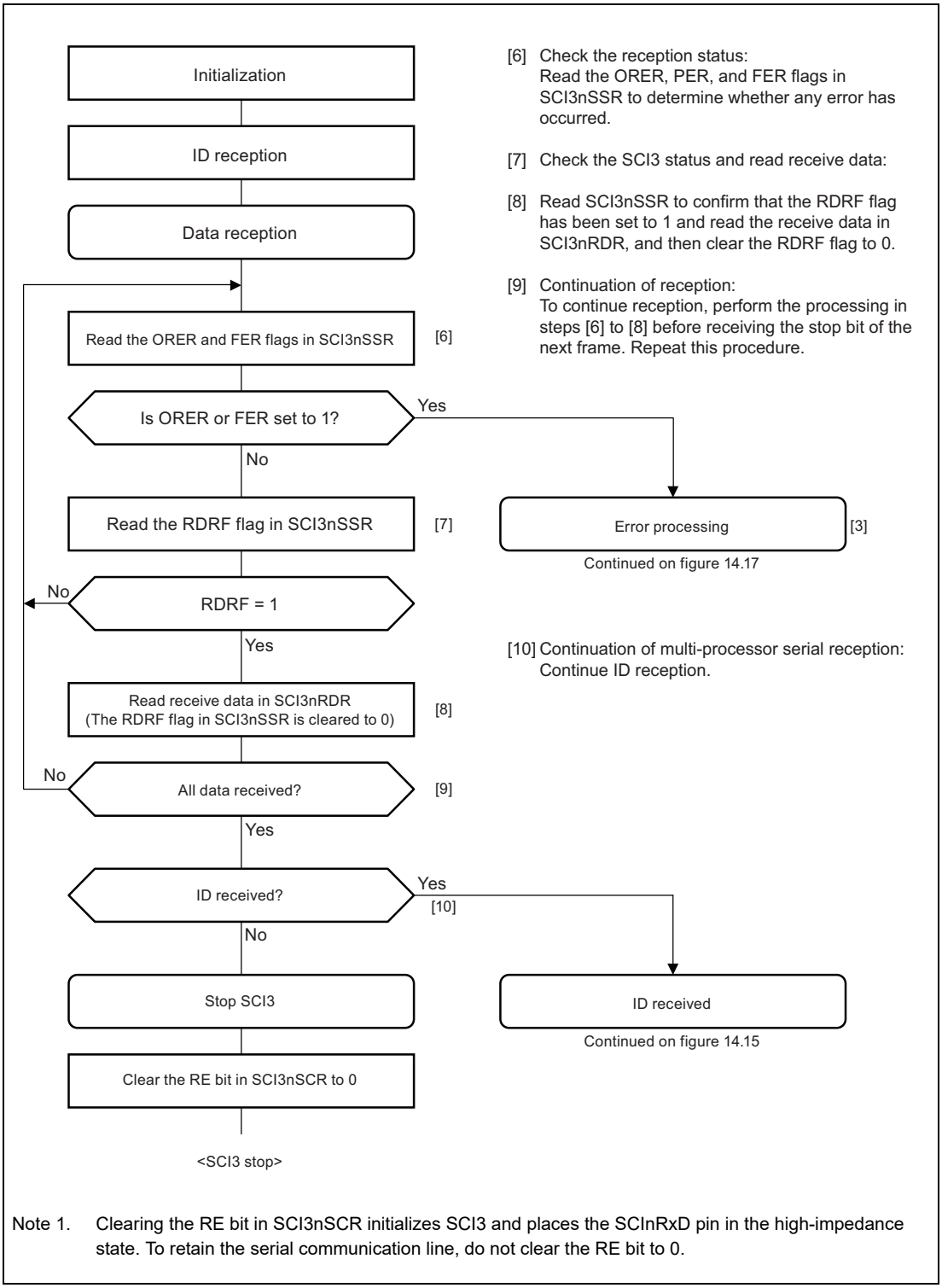
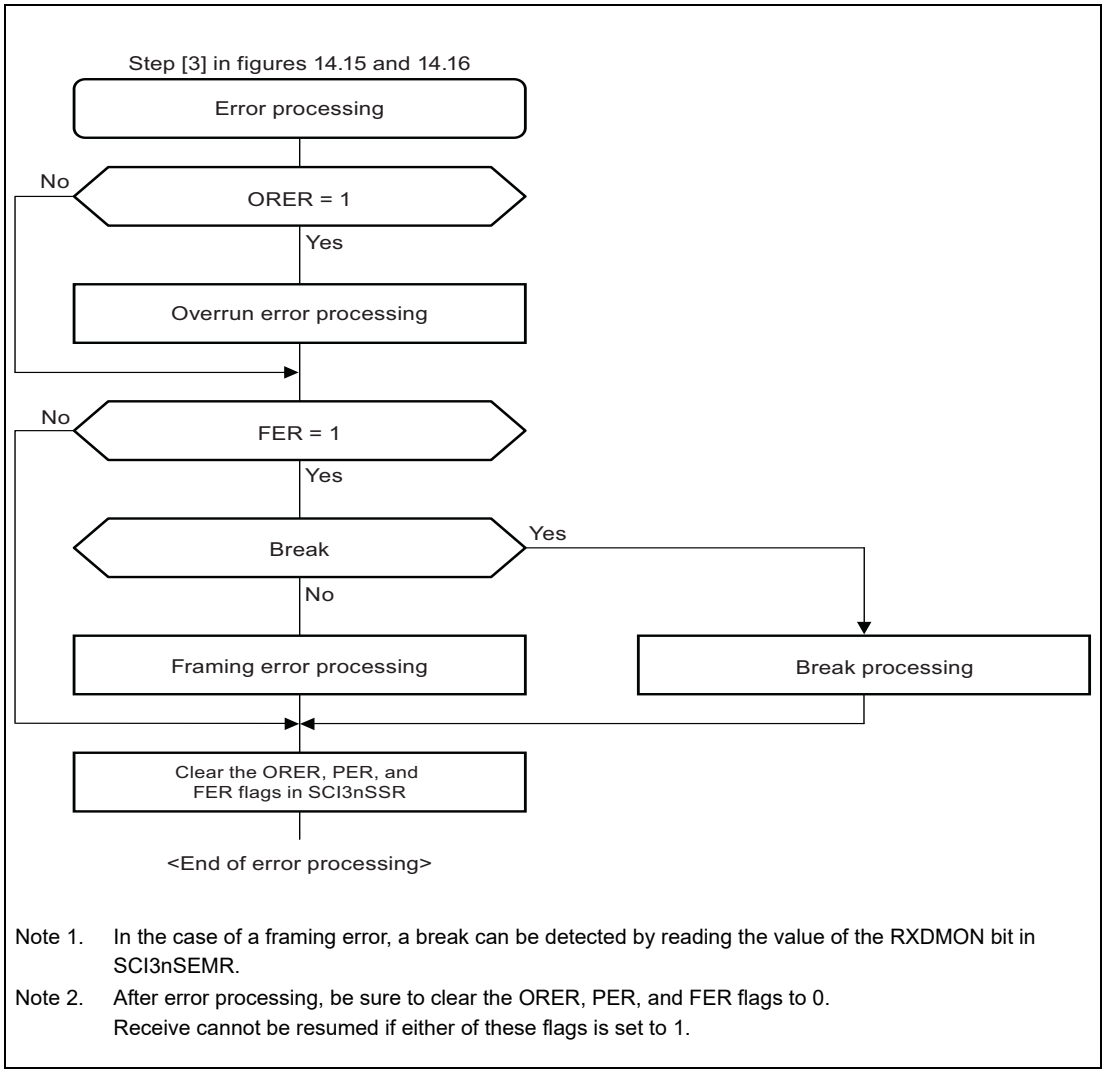


Figure 14.15 Example of Multi-Processor Serial Receive Flowchart (1)



Note 1. Clearing the RE bit in SCI3nSCR initializes SCI3 and places the SCInRxD pin in the high-impedance state. To retain the serial communication line, do not clear the RE bit to 0.

Figure 14.16 Example of Multi-Processor Serial Receive Flowchart (2)



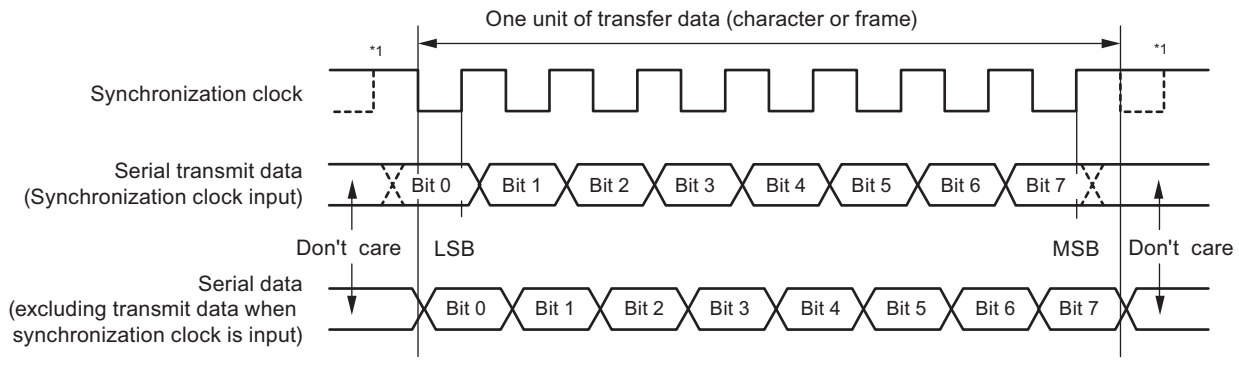
Note 1. In the case of a framing error, a break can be detected by reading the value of the RXDMON bit in SCI3nSEMR.

Note 2. After error processing, be sure to clear the ORER, PER, and FER flags to 0. Receive cannot be resumed if either of these flags is set to 1.

Figure 14.17 Example of Multi-Processor Serial Receive Flowchart (3)

14.6 Operation in Clock Synchronous Mode

Figure 14.18 shows the data format for clock synchronous serial data communication. In clock synchronous mode, data is transmitted or received in synchronization with clock pulses. One character in transfer data consists of 8-bit data. In data transmission with the synchronization clock output, the SCI3 outputs data from one falling edge to the next falling edge of the synchronization clock. In data transmission with the synchronization clock input, the SCI3 outputs the first data (bit 0) after starting transfer immediately after clearing the TDRE bit in SCI3nSSR to 0, and then outputs the next bit data after 2 to 3 ϕ clock cycles from the falling edge of the synchronization clock. In data reception, the SCI3 receives data in synchronization with the rising edge of the synchronization clock. After 8-bit data is output, the communication line holds the last-bit output state. In clock synchronous mode, neither parity bit nor multi-processor bit can be added. The transmitter and the receiver are independent in the SCI3, enabling full-duplex communication by using a common clock. Both the transmitter and the receiver have a double-buffered structure so that the next transmit data can be written during transmission or the previous receive data can be read during reception, enabling continuous data transfer.



Note 1. Holds a high level except during continuous transmission/reception.

Figure 14.18 Data Format in Clock Synchronous Mode (LSB-First)

14.6.1 Clock

An internal clock generated by the on-chip baud rate generator or an external synchronous clock that is input from the SCInSCK pin can be selected by the setting of the CKE1 and CKE0 bits in SCI3nSCR. When operating the SCI3 on the internal clock, a synchronous clock is output from the SCInSCK pin. Eight pulses of the synchronous clock are output during transfer of one character, and the clock is held high while no data is transferred.

14.6.2 SCI3 Initialization (Clock Synchronous Mode)

Before transmitting and receiving data, clear the TE and RE bits in SCI3nSCR to 0 and then initialize the SCI3 according to the sample flowchart in **Figure 14.19**. To switch the operation between transmission, reception, and transmission/reception, clear the TE and RE bits to 0 and then set these bits to the desired value. Before changing the transfer format, be sure to clear the TE and RE bits to 0. Note that clearing the TE bit to 0 sets the TDRE flag to 1, but clearing the RE bit to 0 initializes neither the RDRF, PER, FER, and ORER flags nor SCI3nRDR.

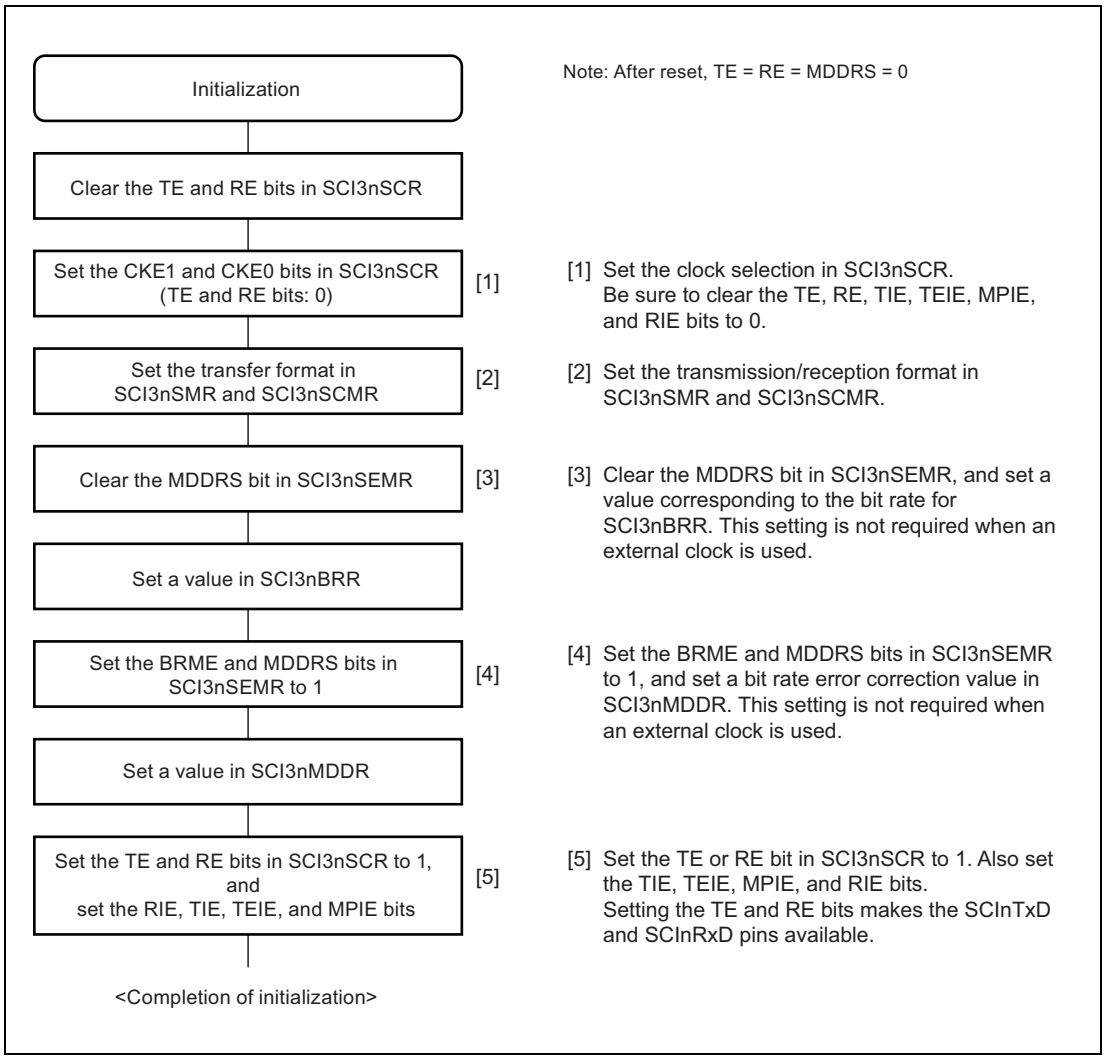


Figure 14.19 Example of SCI3 Initialization Flowchart

14.6.3 Serial Data Transmission (Clock Synchronous Mode)

Figure 14.20 shows an example of the operation for serial transmission in clock synchronous mode. In serial data transmission, the SCI3 operates as described below.

1. When transmit data is written to SCI3nTDR, the TDRE flag is automatically cleared to 0. The SCI3 monitors the TDRE flag in SCI3nSSR. When the flag is cleared, the SCI3 recognizes that data has been written to SCI3nTDR and transfers data from SCI3nTDR to SCI3nTSR, starting output of the first bit when the synchronization clock is input. To write transmit data to SCI3nTDR at a trigger of TXI interrupt request, set the TIE bit to 1 and then set the TE bit to 1 or set both TIE and TE bits simultaneously with one instruction to generate a TXI interrupt request for starting data transfer.
2. Transmission starts after data is transferred from SCI3nTDR to SCI3nTSR, and the TDRE flag is set to 1. When the TIE bit in SCI3nSCR is set to 1 at this time, a TXI interrupt request is generated. Continuous transmission is enabled by writing the next transmit data to SCI3nTDR in this TXI interrupt processing routine before the previously transferred data has been transmitted. When a TEI interrupt request is used, the TIE bit is cleared to 0 after the last transmit data has been written to SCI3nTDR, and the TEIE bit is set to 1.
3. 8-bit data is output from the SCInTxD pin in synchronization with the output clock (when clock output mode has been specified) or in synchronization with the input clock (when external clock has been specified).
4. The SCI3 checks for the TDRE flag at the time of the last bit output.
5. When the TDRE flag is 0, the next transmit data is transferred from SCI3nTDR to SCI3nTSR, and serial transmission of the next frame starts.
6. When the TDRE flag is 1, the TEND flag in SCI3nSSR is set to 1 and the SCInSCK pin retains the output state of the last bit. If the TEIE bit in SCI3nSCR is set to 1 at this time, a TEI interrupt request is generated. The SCInSCK pin is held high.

Figure 14.21 shows a sample flowchart of serial data transmission. Also, **Figure 14.22** shows a sample flowchart for stopping the SCI3 after data transmission. Transmission will not start even if the TDRE flag is cleared while a receive error flag (ORER) is set to 1. Be sure to clear the receive error flags to 0 before starting transmission. Note that clearing the RE bit to 0 does not clear the receive error flags.

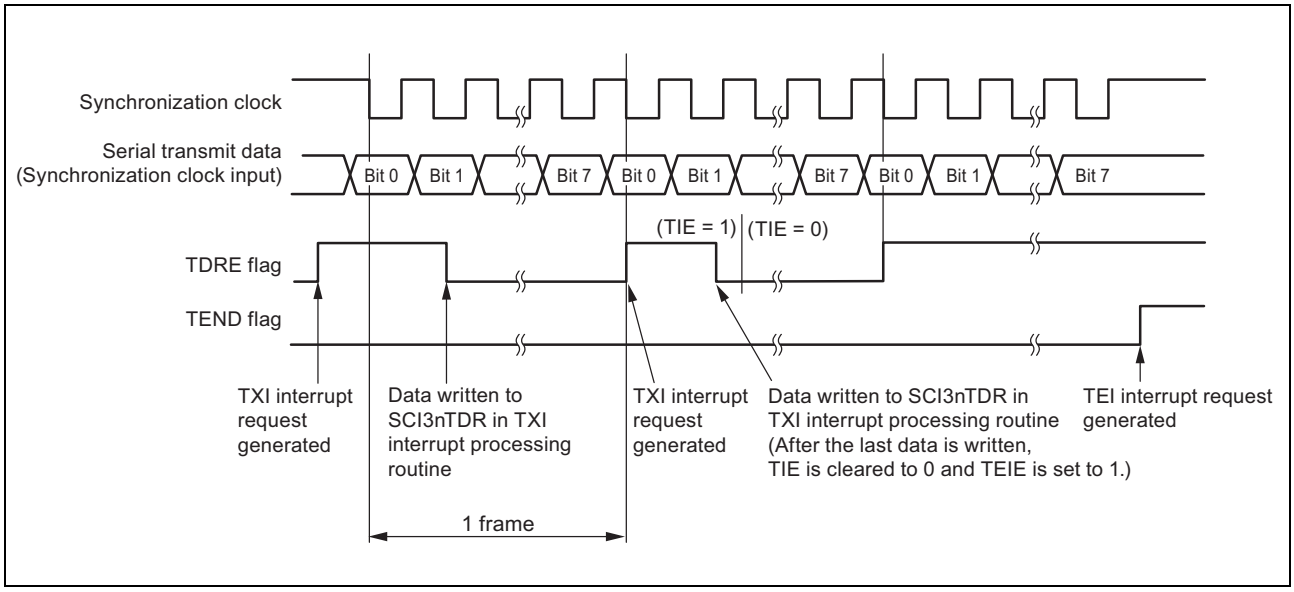


Figure 14.20 Example of Operation for Transmission in Clock Synchronous Mode

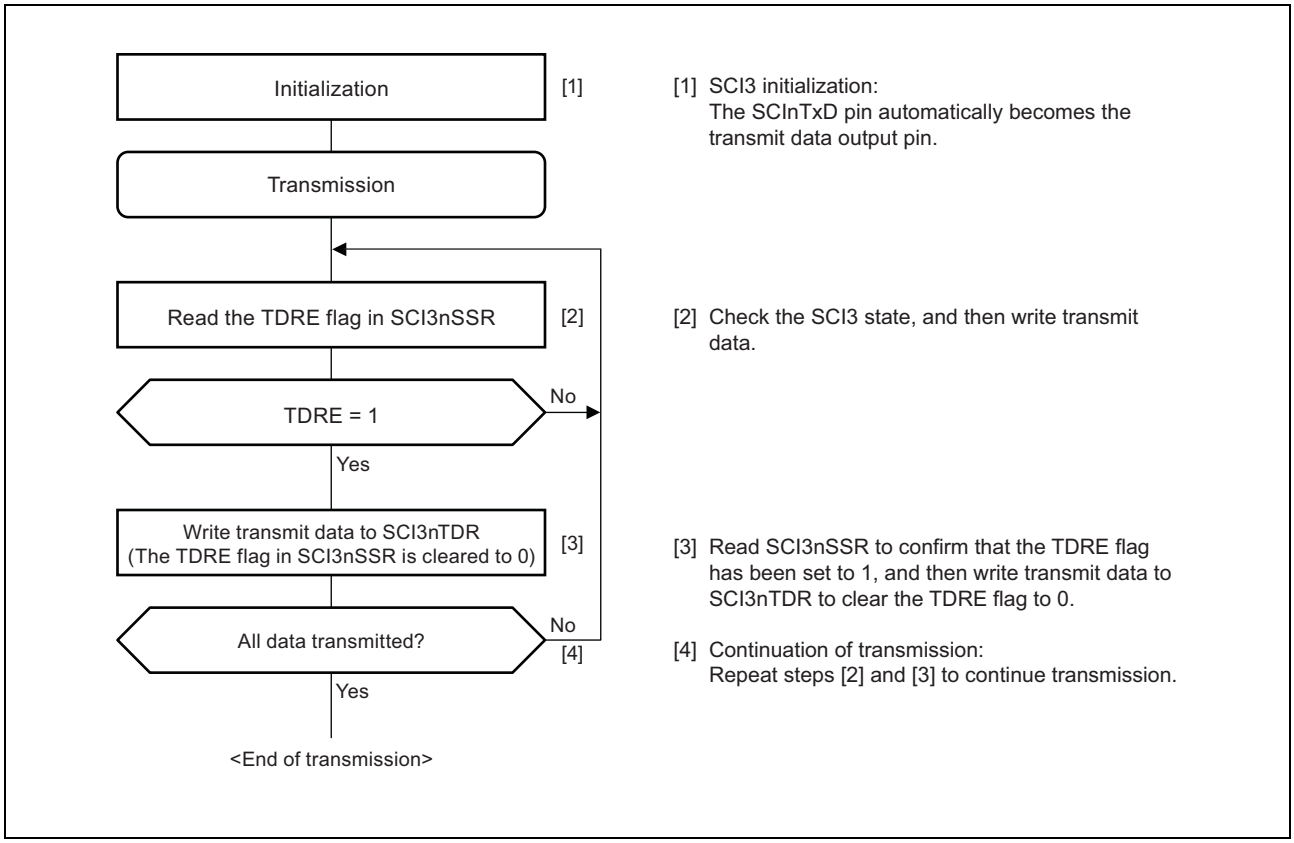
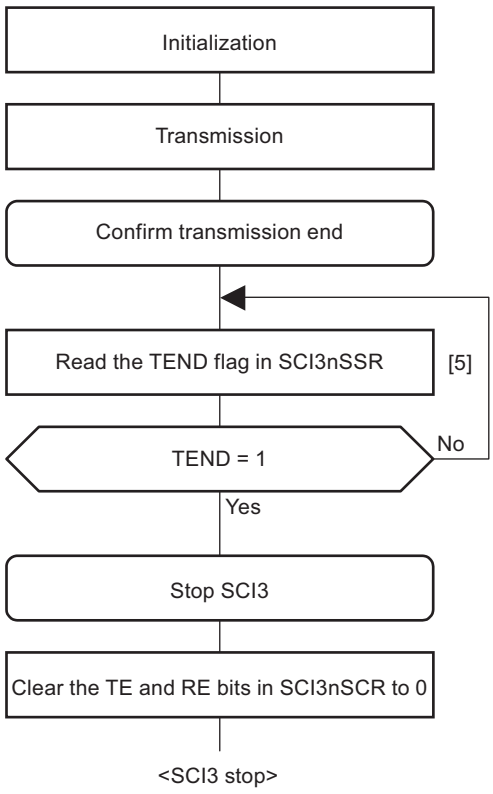


Figure 14.21 Example of Serial Transmission Flowchart



[5] Confirmation of transmission end:
Read the TEND flag in SCI3nSSR to confirm that no data is being transmitted.

Note 1. Clearing the TE and RE bits in SCI3nSCR to 0 initializes SCI3 and places the SCInTxD and SCInRxD pins in the high-impedance state. To retain the serial communications line, switch the SCInTxD pin to port output mode to output the required level before clearing the TE bit to 0. Do not clear the RE bit to 0.

Figure 14.22 Example Flowchart for Stopping the SCI3 after Serial Transmission

14.6.4 Serial Data Receive (Clock Synchronous Mode)

Figure 14.23 shows an example of SCI3 operation for serial reception in clock synchronous mode. In serial data reception, the SCI3 operates as described below.

1. The SCI3 performs internal initialization and starts receiving data in synchronization with a synchronization clock input or output, and stores receive data in SCI3nRSR.
2. When an overrun error occurs (the reception of the next data is completed with the RDRF flag in SCI3nSSR set to 1), the ORER flag in SCI3nSSR is set to 1. If the RIE bit in SCI3nSCR is set to 1 at this time, an ERI interrupt request is generated. Receive data is not transferred to SCI3nRDR. The RDRF flag retains the state of being set to 1.
3. When data has been successfully received, the RDRF flag in SCI3nSSR is set to 1 and the receive data is transferred to SCI3nRDR. When the RIE bit in SCI3nSCR is set to 1 at this time, an RXI interrupt request is generated. Continuous reception is enabled by reading the receive data transferred to SCI3nRDR in this RXI interrupt processing routine before reception of the next data is completed. Reading SCI3nRDR automatically clears the RDRF flag to 0.

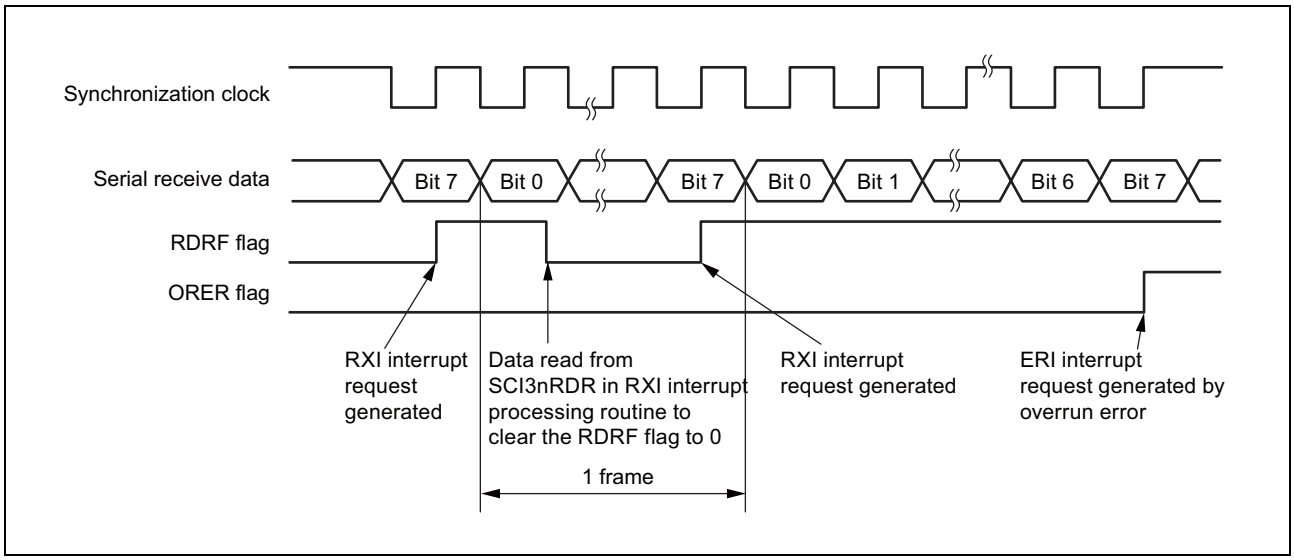
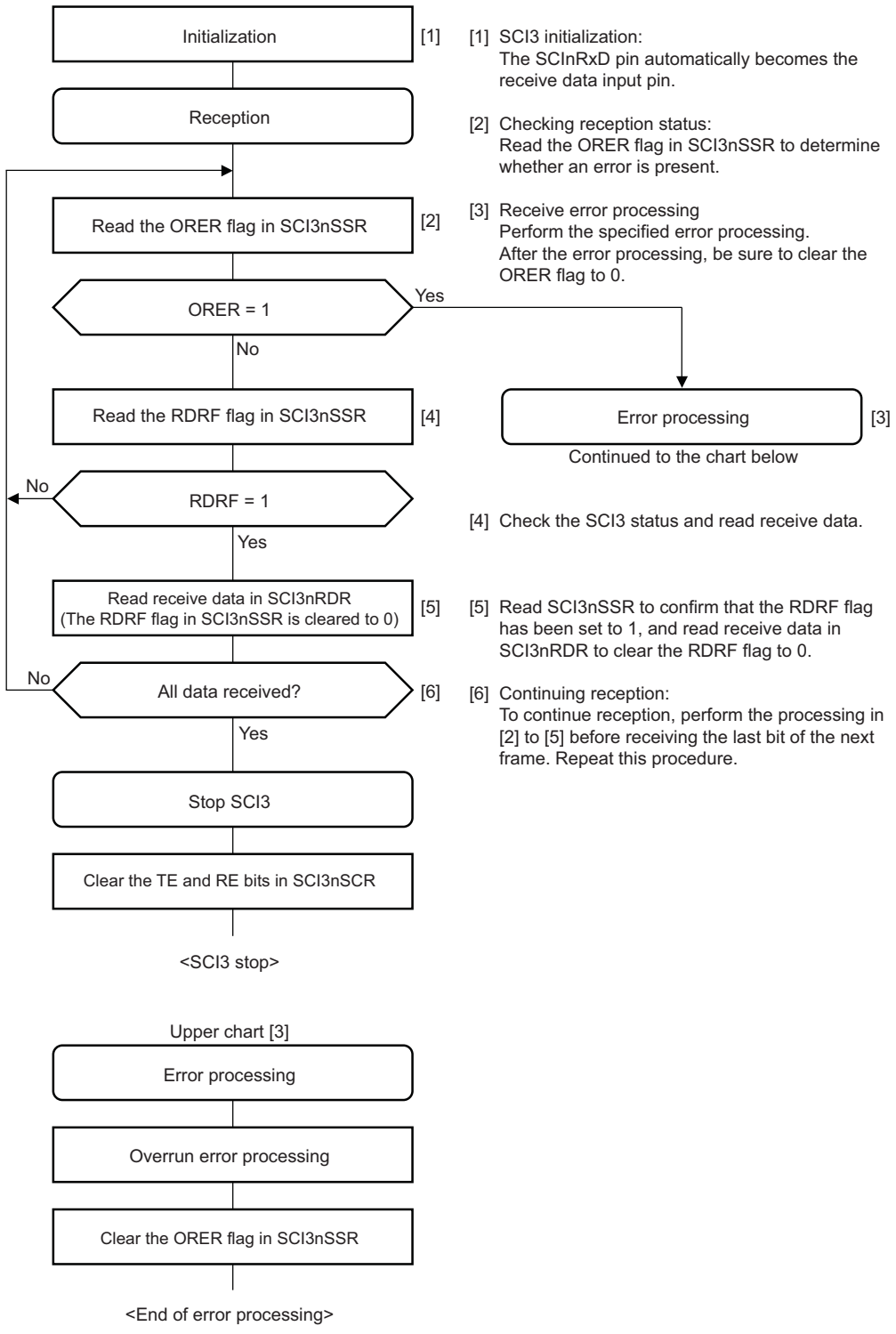


Figure 14.23 Example of SCI3 Operation for Receive

Subsequent transmission and reception are disabled with a receive error flag set to 1. Therefore, be sure to clear the ORER, FER, PER, and RDRF flags before continuing reception. **Figure 14.24** shows an example of flowchart for data reception.



Note 1. Clearing the TE and RE bits in SCI3nSCR initializes the SCI3 and places the SCInTxD and SCInRxD in the high-impedance state. To retain the serial communications line, switch the SCInTxD pin to port output mode to output the required level before clearing the TE bit to 0. Do not clear the RE bit to 0.

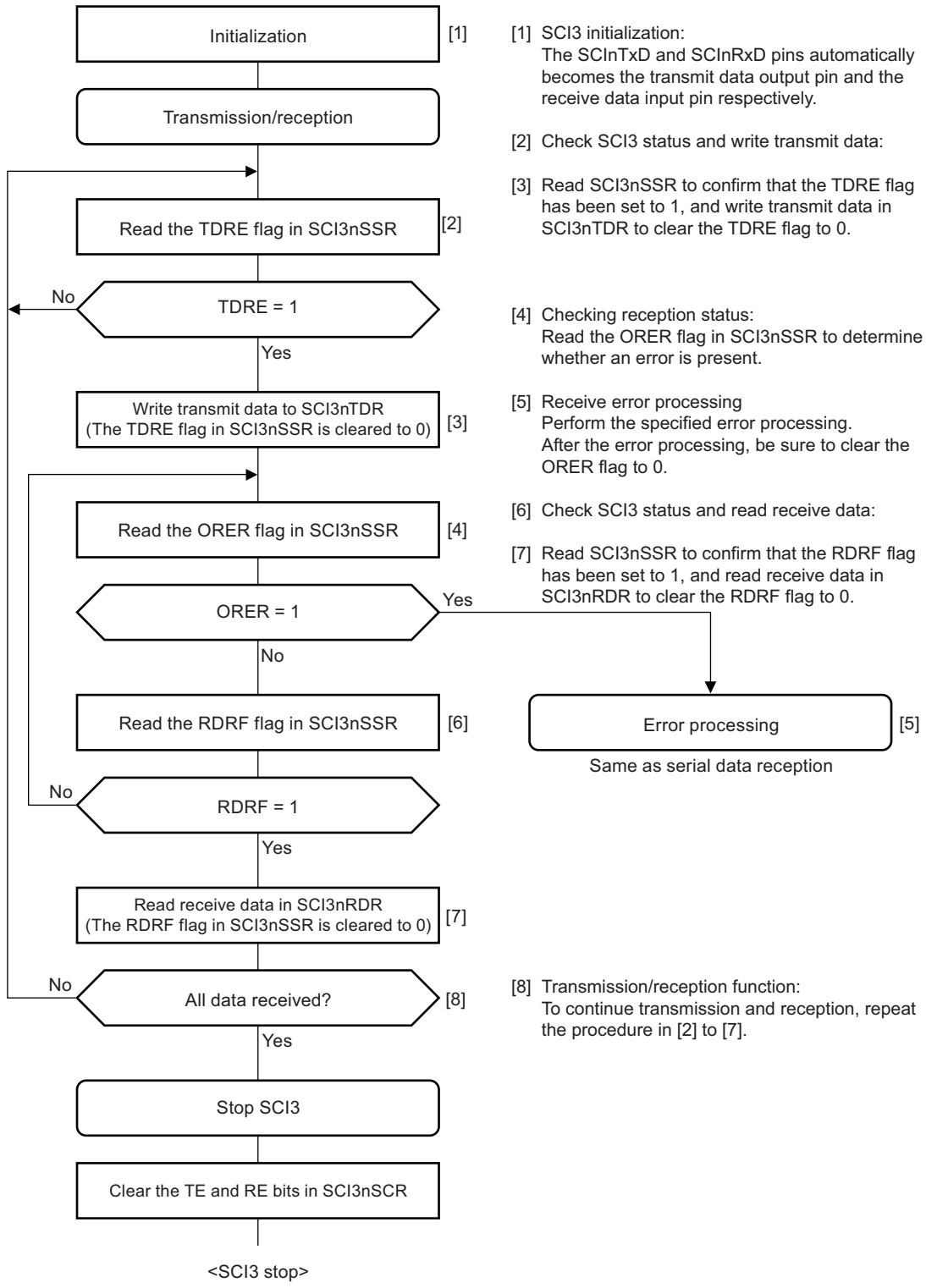
Note 2. After error processing, be sure to clear the ORER, PER, and FER flags to 0. Receive cannot be resumed if either of these flags is set to 1.

Figure 14.24 Example of Serial Receive Flowchart

14.6.5 Simultaneous Serial Data Transmission and Reception (Clock Synchronous Mode)

Figure 14.25 shows a sample flowchart for simultaneous data transmit and receive operations. After the SCI3 is initialized, perform the following procedure for simultaneous data transmit and receive operations.

1. To change transmit mode to simultaneous transmit and receive mode, check that the SCI3 has finished transmission and the TDRE and TEND flags are set to 1. Then clear the TE bit to 0 and then set the TE and RE bits to 1 with a single instruction.
2. To change receive mode to simultaneous transmit and receive mode, check that the SCI3 has finished reception and clear the RE bit to 0. Then check that the RDRF and error flags (ORER, FER, and PER) are cleared to 0 and then set the TE and RE bits to 1 with a single instruction.



Note 1. Clearing the TE and RE bits in SCI3nSCR initializes the SCI3 and places the SCInTxD and SCInRxD pins in the high-impedance state. To retain the serial communications line, switch the SCInTxD pin to port output mode to output the required level before clearing the TE bit to 0. Do not clear the RE bit to 0.

Figure 14.25 Example of Simultaneous Serial Transmission and Receive Flowchart

14.7 Bit Rate Modulation Function

The bit rate modulation function corrects a bit rate by averagely enabling the internal clocks specified by the CKS1 and CKS0 bits in SCI3nSMR for the number specified by SCI3nMDDR out of 256 clocks.

Figure 14.26 shows an example of asynchronous mode in which ϕ clock is selected with the CKS1 and CKS0 bits, SCI3nBRR is set to 0, and SCI3nMDDR is set to 160. In this example, the reference clock cycle is corrected to 256/160 on average and the bit rate is corrected to 160/256. Note that the pulse widths of the internal reference clock expand or contract for the amount of the selected internal clocks because there is a deviation in the enabling of the internal clocks.

Do not use this function with the maximum speed settings (CKS1 and CKS0 bits = 0 in SCI3nSMR, CKE1 bit = 0 in SCI3nSCR, and SCI3nBRR = 0) for clock synchronous mode.

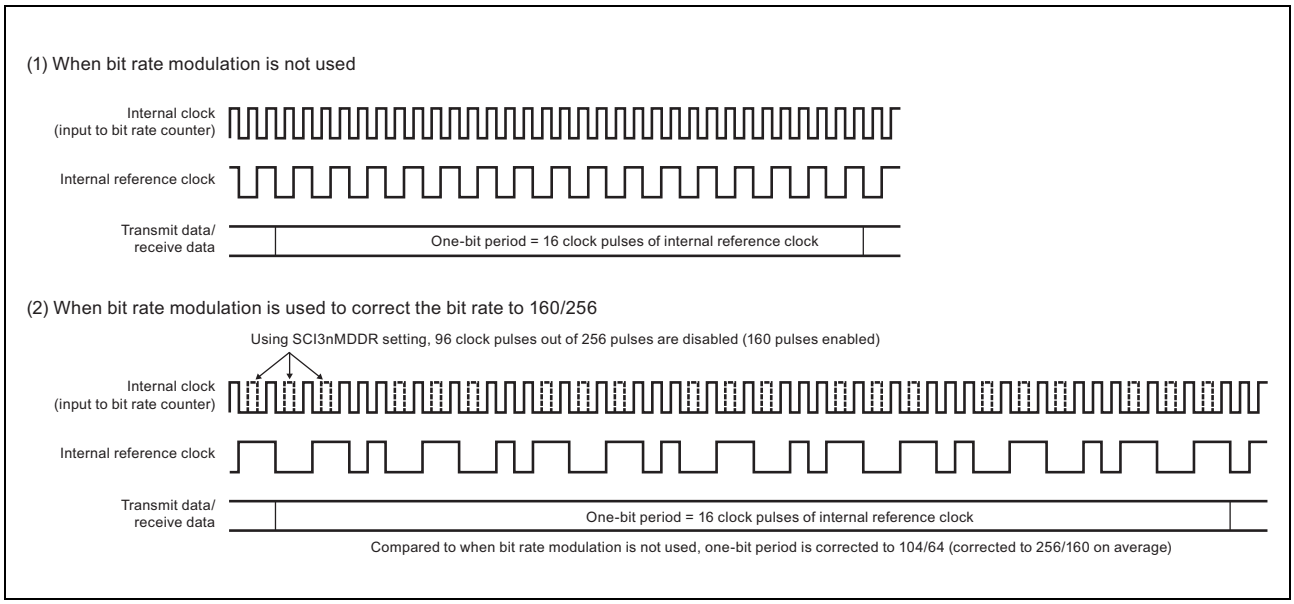


Figure 14.26 Example of Internal Reference Clock when the Bit Rate Modulation Function is Used

14.8 Interrupt Sources

Table 14.20 lists interrupt sources. Each interrupt source outputs an individual interrupt request signal. These interrupt sources can be enabled independently with the enable bit in SCI3nSCR.

A TXI interrupt request is generated when the TDRE flag in SCI3nSSR is set to 1. A TEI interrupt request is generated when the TEND flag in SCI3nSSR is set to 1. A TXI interrupt request can activate the DMAC to handle data transfer. The TDRE flag is automatically cleared to 0 when data is transferred using the DMAC.

CAUTION

The TDRE and TEND flags cannot be cleared to 0 while the TE bit in SCI3nSCR is 0. Since the TEND flag is the level interrupt request flag for a TEI interrupt, do not set the TEIE bit in SCI3nSCR to 1 while the TE bit is 0.

An RXI interrupt request is generated when the RDRF flag in SCI3nSSR is set to 1. An ERI interrupt is generated when either of the ORER, PER, and FER flags in SCI3nSSR is set to 1. An RXI interrupt request can activate the DMAC to handle data transfer. The RDRF flag is automatically cleared to 0 when data is transferred using the DMAC.

A TEI interrupt request is generated when the TEND flag is set to 1 with the TEIE bit set to 1.

CAUTION

If a TEI interrupt request and a TXI interrupt request are generated at the same time, the TXI interrupt request is accepted first. Note that clearing the TDRE flag to 0 at this time in the TXI interrupt processing routine also clears the TEND flag to 0 automatically, disabling the branch to the TEI interrupt processing routine.

Table 14.20 SCI3 Interrupt Sources

Name	Interrupt Source	Interrupt Flag	DMAC Activation	DTS Activation
ERI	Receive error	ORER, FER, PER	Not possible	Not possible
RXI	Receive data full	RDRF	Possible	Not possible
TXI	Transmit data empty	TDRE	Possible	Not possible
TEI	Transmit end	TEND	Not possible	Not possible

14.9 Usage Notes

14.9.1 Break Detection and Processing

A break can be detected by reading the RXDMON bit in SCI3nSEMR when detecting a framing error. Since all inputs from the SCInRxD pin are 0 in the break state, the FER flag is set to 1 and the PER flag may also be set to 1. The SCI3 also continues data reception after it receives a break. For this reason, note that the FER flag is set to 1 again even after the FER flag is cleared to 0.

14.9.2 Mark State and Break Output

While the TE bit is 0 (transmission/reception disabled), the SCInTxD pin can output any level by switching the SCInTxD pin to a general output port. This allows the SCInTxD pin to be the mark state or break output during data transmission.

14.9.3 Receive Error Flags and Transmit Operations in Clock Synchronous Mode

During the clock synchronous simultaneous data transmit/receive operation, transmission cannot be started when a receive error flag (ORER) is set to 1, even if the TDRE flag is cleared to 0. Be sure to clear the receive error flags to 0 before starting transmission. Note that the receive error flags cannot be cleared to 0 even by clearing the RE bit to 0.

14.9.4 Relationship between Writing to SCI3nTDR and the TDRE Flag

The TDRE flag in SCI3nSSR is a status flag indicating that transmit data in SCI3nTDR has been transferred to SCI3nTSR. The TDRE flag is set to 1 when the SCI3 transfers data from SCI3nTDR to SCI3nTSR.

Data can be written to SCI3nTDR regardless of the status of the TDRE flag. However, writing new data to SCI3nTDR while the TDRE flag is 0 will make the data stored in SCI3nTDR lost because it has not been transferred to SCI3nTSR. Therefore, make sure to check that the TDRE flag is set to 1 before writing transmit data to SCI3nTDR.

14.9.5 Restrictions on Using an External Clock for Transmission in Clock Synchronous Mode

When using an external synchronization clock, clear the TDRE flag to 0 and then input a transmission clock (**Figure 14.27**). Also in continuous transmission mode, clear the TDRE flag to 0 and then input a transmission clock for the next frame.

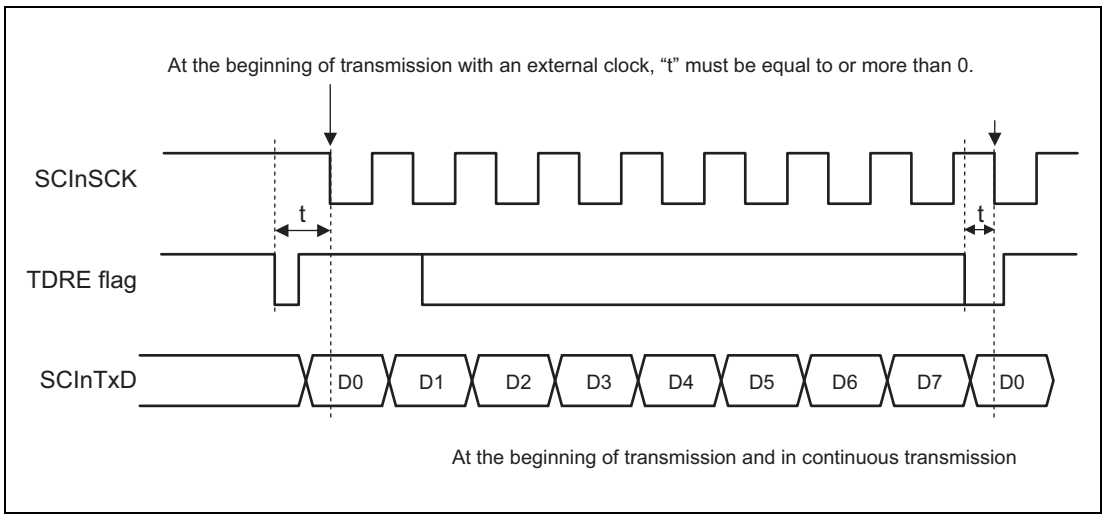


Figure 14.27 Restrictions on Using an External Clock for Transmission in Clock Synchronous Mode

14.9.6 External Clock Input in Clock Synchronous Mode

For the external SCInSCK clock input in clock synchronous mode, see **Section 37.3.7, SCI3/ FLSCI3 Timing**.

Section 15 LIN Master Interface (RLIN2)

This section contains a generic description of the LIN master interface (RLIN2).
The first part of this section describes all RH850/E1M-S specific properties, such as the number of units, register base addresses, etc.

The remainder of the section describes the functions and registers of RLIN2.

15.1 Overview of RH850/E1M-S RLIN2

15.1.1 Units and Channels

This microcontroller has the following number of RLIN2 units and channels.

Table 15.1 Units

Product	RH850/E1x
Units	1
Name	RLIN21n (n = 0)

Table 15.2 Unit Configurations and Channels

Unit Name RLIN21n	Channels per Unit	Unit Channel Number	Channel Name RLIN2m
RLIN210	1	0	RLIN20

Table 15.3 Index

Index	Meaning
n	Throughout this section, the individual RLIN2 units are identified by the index “n” (n = 0).
m	Throughout this section, the individual channels are identified by the index “m” (m = 0).
i	Throughout this section, the individual channels of units that configure RLIN2 are identified by the index “i” (i = 0).
b	Throughout this section, the individual data buffers implemented in RLIN2 are identified by the index “b” (b = 1 to 8).

For example, RLN21nGLWBR are the LIN wake-up baud rate select registers, which are the global registers of RLN21nmLiMD are the LIN mode registers, which are the channel registers.

15.1.2 Register Base Addresses

RLIN2 base addresses are listed in the following table.

RLIN2 register addresses are given as offsets from the base addresses in general.

Table 15.4 Register Base Addresses

Base Address Name	Base Address
<RLIN210_base>	FFCE 0000_H

15.1.3 Clock Supply

RLIN2 clock is listed in following table.

Table 15.5 Clock Supply

Unit Name	Clock for the Unit	Supply Clock Name
RLIN21n	pclk (P-bus clock)	CLK_LSB (low-speed peripheral clock)
	clkc (LIN communication clock source)	CLKC_LSB (unmodulated low-speed peripheral clock)

15.1.4 Interrupt Request

RLIN2 interrupt requests are listed in the following table.

Table 15.6 Interrupt Requests

Unit Interrupt Name	Outline	Interrupt Number
INTRLIN0	Transmission complete interrupt/ reception complete interrupt/ error detection interrupt (RLIN210 interrupt)	429

15.1.5 Reset Sources

RLIN2 reset sources are listed in the following table. RLIN2 is initialized by these reset sources.

Table 15.7 Reset Sources

Unit Name	Reset Source
RLIN21n	Any reset source

15.1.6 External Input/Output Signals

External input/output signals of RLIN2 are listed below.

Table 15.8 I/O Signals of RLIN2n

Unit Signal Name	Outline	Alternative Port Pin Signal
RLIN210		
RLIN20RX	RLIN210 receive data input	LRX0
RLIN20TX	RLIN210 transmit data output	LTX0

15.2 Function

15.2.1 Functional Overview

The LIN master interface is a hardware LIN communication controller that complies to LIN Specification Package Revision 1.3, 2.0, 2.1, 2.2, and SAEJ2602 (SEP 2005), and automatically performs frame communication and error determination.

Table 15.9 gives the LIN master interface specifications and **Figure 15.1** shows a block diagram of the LIN master interface.

Table 15.9 LIN Master Interface Specifications

Item		Specifications
Channel count		1 channel
LIN communication function	Protocol	LIN Specification Package Revision 1.3, 2.0, 2.1, 2.2, and SAEJ2602 (SEP 2005)
	Variable frame structure	• Break transmission width: 13 to 28 Tbits • Break delimiter transmission width: 1 to 4 Tbits • Inter-byte space (header): 0 to 7 Tbits (space between Sync field and ID field)*1 • Response space: 0 to 7 Tbits*1 • Inter-byte space: 0 to 3 Tbits (space between data bytes in response area) • Transmission wake-up: 1 to 16 Tbits
	Checksum	• Automatic operation for both transmission and reception • Classic or enhanced selectable (for each frame)
	Response field data byte count	Variable from 0 to 8 bytes
	Frame communication modes	• Mode in which header transmission and response transmission/reception is started with a single transmission start request • Mode in which header transmission and response transmission are started with separate transmission start requests (frame separate mode)
	Wake-up transmission and reception	LIN wake-up mode provided • Wake-up transmission (1 to 16 Tbits) • Wake-up reception Low-level width of input signals measured
	Status	• Successful frame/wake-up transmission • Successful header transmission • Successful frame/wake-up reception*2 • Successful data 1 reception • Error detection • Operation mode (LIN reset mode, LIN wake-up mode, LIN operation mode, LIN self-test mode)
	Error status	• Bit error • Checksum error • Frame timeout error • Physical bus error • Framing error
	Baud rate selection	Baud rate conforming to the LIN specifications generated using baud rate generator
	Test mode	Self-test mode for user evaluation
	Interrupt function	• Successful frame/wake-up transmission • Successful frame/wake-up reception*2 • Error detection These three logical OR of these three operations events is are the interrupt sources (INTRLINm) for each channel.

Note 1. Since the same register is used for setting, the inter-byte space (header) = response space.

Note 2. For wake-up reception, the low level width of the input signal is indicated.

15.2.2 Block Diagram

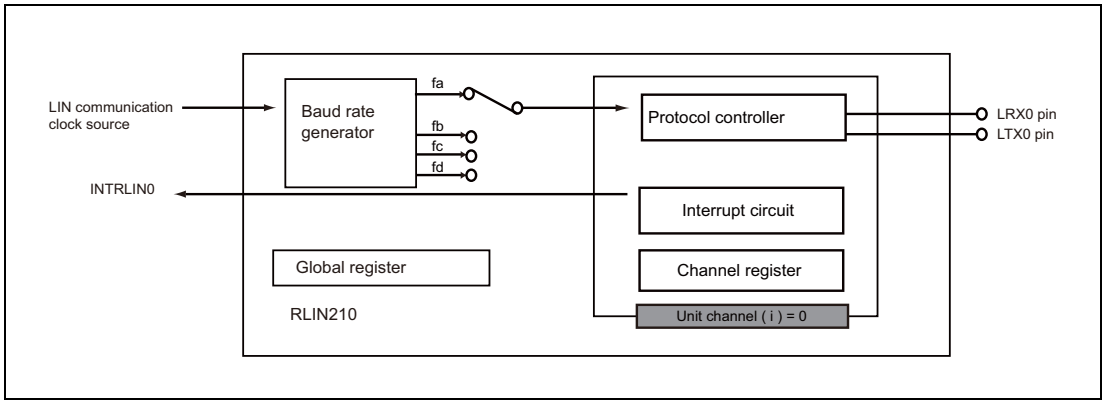


Figure 15.1 LIN Master Interface Block Diagram

15.3 Registers

The registers of the LIN master interface are configured with global registers and channel registers. As the global registers are allocated for each unit, they can be set respectively in units. As the channel registers are allocated for each channel, they can be controlled respectively in channels.

15.3.1 List of Registers

RLIN2 registers are listed in the following table.

For information on <RLIN21n_base>, see **Section 15.1.2, Register Base Addresses**.

Table 15.10 LIN Master Interface Registers Overview

Module name	Register name	Shortcut	Address
Global registers			
RLN21nG	LIN wake-up baud rate select register	RLN21nGLWBR	<RLIN21n_base> + 01_H
RLN21nG	LIN baud rate prescaler 0 register	RLN21nGLBRP0	<RLIN21n_base> + 02_H
RLN21nG	LIN baud rate prescaler 1 register	RLN21nGLBRP1	<RLIN21n_base> + 03_H
RLN21nG	LIN self test control register	RLN21nGLSTC	<RLIN21n_base> + 04_H
Channel registers			
RLN21nm	LIN mode register	RLN21nmLiMD	<RLIN21n_base> + 08_H
RLN21nm	LIN break field configuration register	RLN21nmLiBFC	<RLIN21n_base> + 09_H
RLN21nm	LIN space configuration register	RLN21nmLiSC	<RLIN21n_base> + $0A_H$
RLN21nm	LIN wake-up configuration register	RLN21nmLiWUP	<RLIN21n_base> + $0B_H$
RLN21nm	LIN interrupt enable register	RLN21nmLiIE	<RLIN21n_base> + $0C_H$
RLN21nm	LIN error detection enable register	RLN21nmLiEDE	<RLIN21n_base> + $0D_H$
RLN21nm	LIN control register	RLN21nmLiCUC	<RLIN21n_base> + $0E_H$
RLN21nm	LIN transmission control register	RLN21nmLiTRC	<RLIN21n_base> + 10_H
RLN21nm	LIN mode status register	RLN21nmLiMST	<RLIN21n_base> + 11_H
RLN21nm	LIN status register	RLN21nmLiST	<RLIN21n_base> + 12_H
RLN21nm	LIN error status register	RLN21nmLiEST	<RLIN21n_base> + 13_H
RLN21nm	LIN data field configuration register	RLN21nmLiDFC	<RLIN21n_base> + 14_H
RLN21nm	LIN ID buffer register	RLN21nmLiIDB	<RLIN21n_base> + 15_H
RLN21nm	LIN check sum buffer register	RLN21nmLiCBR	<RLIN21n_base> + 16_H
RLN21nm	LIN data buffer 1 register	RLN21nmLiDBR1	<RLIN21n_base> + 18_H
RLN21nm	LIN data buffer 2 register	RLN21nmLiDBR2	<RLIN21n_base> + 19_H
RLN21nm	LIN data buffer 3 register	RLN21nmLiDBR3	<RLIN21n_base> + $1A_H$
RLN21nm	LIN data buffer 4 register	RLN21nmLiDBR4	<RLIN21n_base> + $1B_H$
RLN21nm	LIN data buffer 5 register	RLN21nmLiDBR5	<RLIN21n_base> + $1C_H$
RLN21nm	LIN data buffer 6 register	RLN21nmLiDBR6	<RLIN21n_base> + $1D_H$
RLN21nm	LIN data buffer 7 register	RLN21nmLiDBR7	<RLIN21n_base> + $1E_H$
RLN21nm	LIN data buffer 8 register	RLN21nmLiDBR8	<RLIN21n_base> + $1F_H$

15.3.2 Global Registers

15.3.2.1 RLN21nGLWBR — LIN Wake-up Baud Rate Select Register

Access: This register can be read/written in 8-bit units.

Address: <RLIN21n_base> + 01_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	LWBR0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 15.11 RLN21nGLWBR Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	The write value should be the value after reset.
0	LWBR0	Wake-up Baud Rate Select 0: In LIN wake-up mode, the clock specified by the LCKS bit setting in the RLN21nmLiMD register is used. (when LIN1.3 is used) 1: In LIN wake-up mode, the clock fa is used regardless of the setting of the LCKS bit of the RLN21nmLiMD register. (When LIN2.x is used)

Set the RLN21nGLWBR register when all channels in the same unit are in LIN reset mode (while the OMM0 bit in the RLN21nmLiMST register is 0).

LWBR0 Bit (Wake-up Baud Rate Select)

When LIN Specification Package Revision 1.3 is used, set the LWBR0 bit in the RLN21nGLWBR register to 0. This allows the 2.5-Tbit or longer low-level width of the input signal to be measured. When LIN Specification Package Revision 2.x is used, set the LWBR0 bit to 1.

With this setting, fa is selected as the LIN system clock (fLIN) during LIN wake-up mode regardless of the setting of the LCKS bit in the RLN21nmLiMD register (the LCKS bit is not changed) and the 2.5-Tbit or longer low-level width of the input signal to be measured.

Setting the baud rate to 19200 bps while fa is selected allows the 130µs or longer low-level width of the input signal to be detected during LIN wake-up mode regardless of the setting of the LCKS bit in the RLN21nmLiMD register.

15.3.2.2 RLN21nGLBRP0 — LIN Baud Rate Prescaler 0 Register

Access: This register can be read/written in 8-bit units

Address: <RLIN21n_base> + 02_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	LBRP0[7:0]							
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 15.12 RLN21nGLBRP0 Register Contents

Bit Position	Bit Name	Function
7 to 0	LBRP0[7:0]	Assuming that the value set in this register is N (0 to 255), the baud rate prescaler 0 divides the frequency of the LIN communication clock source by N + 1. Setting Range: 00_H to FF_H

Set the RLN21nGLBRP0 register when all channels in the same unit are in LIN reset mode (while the OMM0 bit in the RLN21nmLiMST register is 0).

The value set in this register is used to control the frequency of baud rate clock source fa, fb, and fc.

Assuming that the value set in this register is N, the baud rate prescaler 0 divides the frequency of the LIN communication clock source by N+1.

15.3.2.3 RLN21nGLBRP1 — LIN Baud Rate Prescaler 1 Register

Access: This register can be read/written in 8-bit units.

Address: <RLIN21n_base> + 03_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	LBRP1[7:0]							
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 15.13 RLN21nGLBRP1 Register Contents

Bit Position	Bit Name	Function
7 to 0	LBRP1[7:0]	Assuming that the value set in this register is M (0 to 255), the baud rate prescaler 1 divides the frequency of the LIN communication clock source by M+1. Setting Range: 00_H to FF_H

Set the RLN21nGLBRP1 register when all channels in the same unit are in LIN reset mode (while the OMM0 bit in the RLN21nmLiMST register is 0).

The value set in this register is used to control the frequency of baud rate clock source fd.

Assuming that the value set in this register is M, the baud rate prescaler 1 divides the frequency of the LIN communication clock source by M+1.

15.3.2.4 RLN21nGLSTC — LIN Self-Test Control Register

This register is used to unlock protection of LIN self-test mode.

Set the RLN21nGLSTC register when all channels in the same unit are while the OMM0 bit in the RLN21nmLiMST register is 0 (in LIN reset mode).

For making transition to LIN self-test mode, refer to **Section 15.15, LIN Self-Test Mode**.

Access: This register can be read/written in 8-bit units.

Address: <RLIN21n_base> + 04_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	LSTM[7:0]							
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1

Note 1. Writing is ignored in modes other than LIN reset mode.

Table 15.14 RLN21nGLSTC Register Contents

Bit Position	Bit Name	Function
7 to 0	LSTM	LIN Self-Test Mode 0: The module is not in LIN self-test mode*1 1: The module is in LIN self-test mode.*2 Writing $A7_H$, 58_H, and 01_H successively to the RLN21nGLSTC register places the module into LIN self-test mode. Setting other than the above is prohibited. If any other value is set, the read value is undefined.

Note 1. For leaving LIN self-test mode, see **Section 15.15.4, Terminating LIN Self-Test Mode**. When LIN self-test mode is terminated, 00_H is read.

Note 2. See **Section 15.15.1, Change to LIN Self-Test Mode** for how to change to LIN self-test mode. When transition to LIN self-test mode is completed, 01_H is read.

15.3.3 Channel Registers

15.3.3.1 RLN21nmLiMD — LIN Mode Register

Access: This register can be read/written in 8-bit units.

Address: RLN21nmLiMD: <RLIN21n_base> + 08_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	LCKS[1:0]		—	—
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R	R

Table 15.15 RLN21nmLiMD Register Contents

Bit Position	Bit Name	Function
7 to 4	Reserved	The write value should be the value after reset.
3, 2	LCKS[1:0]	LIN System Clock Select b3 b2 0 0: fa (Clock generated by baud rate prescaler 0) 0 1: fb (1/2 clock generated by baud rate prescaler 0) 1 0: fc (1/8 clock generated by baud rate prescaler 0) 1 1: fd (1/2 clock generated by baud rate prescaler 1)
1, 0	Reserved	These bits are always read as 0. The write value should always be 0.

Set the RLN21nmLiMD register while the OMM0 bit in the RLN21nmLiMST register is 0 (in LIN reset mode).

LCKS[1:0] Bits (LIN System Clock Select)

The LCKS bits select the clock to be input to the protocol controller.
With 00_B set, the protocol controller is provided with fa (clock generated by baud rate prescaler 0).
With 01_B set, the protocol controller is provided with fb (1/2 clock generated by baud rate prescaler 0).
With 10_B set, the protocol controller is provided with fc (1/8 clock generated by baud rate prescaler 0).
With 11_B set, the protocol controller is provided with fd (1/2 clock generated by baud rate prescaler 1).
When the LWBR0 bit in the RLN21nGLWBR is 1 (when LIN 2.x is used) and the RLN21nmLiMST register is 01_H (LIN wake-up mode), regardless of the setting of the LWBR0 bit, fa is input to the protocol controller (LCKS bit is not changed).

15.3.3.2 RLN21nmLiBFC — LIN Break Field Configuration Register

Access: This register can be read/written in 8-bit units.

Address: RLN21nmLiBFC: <RLIN21n_base> + 09_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	BDT[1:0]		BLT[3:0]			
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Table 15.16 RLN21nmLiBFC Register Contents

Bit Position	Bit Name	Function
7, 6	Reserved	The write value should be the value after reset.
5, 4	BDT[1:0]	Transmission Break Delimiter (High Level) Width Setting b5 b4 0 0: 1 Tbit 0 1: 2 Tbits 1 0: 3 Tbits 1 1: 4 Tbits
3 to 0	BLT[3:0]	Transmission Break (Low Level) Width Setting b3 b0 0 0 0 0: 13 Tbits 0 0 0 1: 14 Tbits 0 0 1 0: 15 Tbits : 1 1 1 0: 27 Tbits 1 1 1 1: 28 Tbits

Set the RLN21nmLiBFC register while the OMM0 bit in the RLN21nmLiMST register is 0 (in LIN reset mode).

Some combinations of the set values result in the length of a frame exceeding the timeout time. Set the appropriate values in this register.

BDT[1:0] Bits (Transmission Break Delimiter (High Level) Width Setting)

This bit is used to set the break high level width of transmission frame header.
1 Tbit to 4 Tbits can be set.

BLT[3:0] Bits (Transmission Break (Low Level) Width Setting

This BLT bits set the break low level width of transmission frame header.
13 Tbits to 28 Tbits can be set.

15.3.3.3 RLN21nmLiSC — LIN Space Configuration Register

Access: This register can be read/written in 8-bit units.

Address: RLN21nmLiSC: <RLIN21n_base> + $0A_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	IBS[1:0]		—	IBHS[2:0]		
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R	R/W	R/W	R/W

Table 15.17 RLN21nmLiSC Register Contents

Bit Position	Bit Name	Function
7, 6	Reserved	The write value should be the value after reset.
5, 4	IBS[1:0]	Inter-Byte Space Select b5 b4 0 0: 0 Tbit 0 1: 1 Tbit 1 0: 2 Tbits 1 1: 3 Tbits
3	Reserved	The write value should be the value after reset.
2 to 0	IBHS[2:0]	Inter-Byte Space (Header)/Response Space Select b2 b0 0 0 0: 0 Tbit 0 0 1: 1 Tbit 0 1 0: 2 Tbits 0 1 1: 3 Tbits 1 0 0: 4 Tbits 1 0 1: 5 Tbits 1 1 0: 6 Tbits 1 1 1: 7 Tbits

Set the RLIN21nmLiSC register while the OMM0 bit in the RLN21nmLiMST register is 0 (in LIN reset mode).

Some combinations of the set values result in the length of a frame or a response exceeding the timeout time. Set the appropriate values in this register.

IBS[1:0] Bits (Inter-Byte Space Select)

The IBS bits set the width of the inter-byte space of the transmission frame response field.
0 Tbit to 3 Tbits can be set.
These bits are enabled only during response transmission; these are disabled during response reception.

IBHS[2:0] Bits (Inter-Byte Space (Header)/Response Space Select)

The IBHS bits set the width of the inter-byte space (header) of the transmission frame header field and the response space.
0 Tbit to 7 Tbits can be set.
The response space setting is enabled only during response transmission; setting is disabled during response reception.
The inter-byte space (header) is equal to the response space.

15.3.3.4 RLN21nmLiWUP — LIN Wake-up Configuration Register

Access: This register can be read/written in 8-bit units.

Address: RLN21nmLiWUP: <RLIN21n_base> + $0B_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	WUTL[3:0]				—	—	—	—
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R	R	R

Table 15.18 RLN21nmLiWUP Register Contents

Bit Position	Bit Name	Function
7 to 4	WUTL[3:0]	Wake-up Transmission Low level Width Select b7 b4 0 0 0 0: 1 Tbit 0 0 0 1: 2 Tbits 0 0 1 0: 3 Tbits 0 0 1 1: 4 Tbits : 1 1 0 0: 13 Tbits 1 1 0 1: 14 Tbits 1 1 1 0: 15 Tbits 1 1 1 1: 16 Tbits
3 to 0	Reserved	The write value should be the value after reset.

Set the RLN21nmLiWUP register while the OMM0 bit in the RLN21nmLiMST register is 0 (in LIN reset mode).

WUTL[3:0] Bits (Wake-up Transmission Low level Width Select)

The WUTL bits set the low level width of the wake-up signal transmission.
1 Tbit to 16 Tbits can be set.

When the LWBR0 bit in the RLN21nGLWBR is 1 (when LIN 2.x is used), regardless of the setting of the the LCKS bit in the RLN21nmLiMD register, fa is selected as the LIN system clock (fLIN) (LCKS bit is not changed).

15.3.3.5 RLN21nmLiIE — LIN Interrupt Enable Register

Access: This register can be read/written in 8-bit units.

Address: RLN21nmLiIE: <RLIN21n_base> + $0C_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	ERRIE	FRCIE	FTCIE
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W

Table 15.19 RLN21nmLiIE Register Contents

Bit Position	Bit Name	Function
7 to 3	Reserved	The write value should be the value after reset.
2	ERRIE	Error Detection Interrupt Request Enable 0: Disables error detection interrupt request. 1: Enables error detection interrupt request.
1	FRCIE	Successful Frame/Wake-up Reception Interrupt Request Enable 0: Disables successful frame/wake-up reception interrupt request. 1: Enables successful frame/wake-up reception interrupt request.
0	FTCIE	Successful Frame/Wake-up Transmission Interrupt Request Enable 0: Disables successful frame/wake-up transmission interrupt request. 1: Enables successful frame/wake-up transmission interrupt request.

Set the RLN21nmLiIE register while the OMM0 bit in the RLN21nmLiMST register is 0 (in LIN reset mode).

ERRIE Bit (Error Detection Interrupt Request Enable)

The ERRIE bit enables or disables interrupt request generation upon detection of an error.
With 0 set, the interrupt request is not generated when the ERR flag in the RLN21nmLiST register is set to 1.
With 1 set, the interrupt request is generated when the ERR flag in the RLN21nmLiST register is set to 1.
Error sources can be the bit error, physical bus error, frame timeout error, framing error, and checksum error.
Detection of the bit error, physical bus error, frame/response timeout error, and framing error can be enabled or disabled using the RLN21nmLiEDE register.

FRCIE Bit (Successful Frame/Wake-up Reception Interrupt Request Enable)

The FRCIE bit enables or disables interrupt request generation upon successful reception of a frame or a wake-up signal (counting of low level width of the input signal).
With 0 set, the interrupt request is not generated when the FRC flag in the RLN21nmLiST register is set to 1.
With 1 set, the interrupt request is generated when the FRC flag in the RLN21nmLiST register is set to 1.

FTCIE Bit (Successful Frame/Wake-up Transmission Interrupt Request Enable)

The FTCIE bit enables or disables interrupt request generation upon successful transmission of a frame or a wake-up signal.
With 0 set, the interrupt request is not generated when the FTC flag in the RLN21nmLiST register is set to 1.
With 1 set, the interrupt request is generated when the FTC flag in the RLN21nmLiST register is set to 1.

15.3.3.6 RLN21nmLiEDE — LIN Error Detection Enable Register

Access: This register can be read/written in 8-bit units

Address: RLN21nmLiEDE: <RLIN21n_base> + $0D_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	FERE	FTERE	PBERE	BERE
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 15.20 RLN21nmLiEDE Register Contents

Bit Position	Bit Name	Function
7 to 4	Reserved	The write value should be the value after reset.
3	FERE	Framing Error Detection Enable 0: Disables framing error detection. 1: Enables framing error detection.
2	FTERE	Frame Timeout Error Detection Enable 0: Disables frame timeout error detection. 1: Enables frame timeout error detection.
1	PBERE	Physical Bus Error Detection Enable 0: Disables physical bus error detection. 1: Enables physical bus error detection.
0	BERE	Bit Error Detection Enable 0: Disables bit error detection. 1: Enables bit error detection.

Set the RLN21nmLiEDE register while the OMM0 bit in the RLN21nmLiMST register is 0_B (in LIN reset mode).

FERE Bit (Framing Error Detection Enable)

The FERE bit enables or disables detection of the framing error.
With 0 set, the framing error is not detected.
With 1 set, the framing error is detected.
When this bit is set to 1, the detection result is indicated in the FER flag in the RLN21nmLiEST register.
For details of the framing error, refer to **Section 15.14, Error Status**.

FTERE Bit (Frame Timeout Error Detection Enable)

The FTERE bit enables or disables detection of the frame timeout error.
With 0 set, the frame timeout error is not detected.
With 1 set, the frame timeout error is detected.
When this bit is set to 1, the detection result is indicated in the FTER flag in the RLN21nmLiEST register.
Do not use the timeout error if response data of 9 bytes or more is to be transmitted or received.
For details of the frame timeout error, refer to **Section 15.14, Error Status**.

PBERE Bit (Physical Bus Error Detection Enable)

The PBERE bit enables or disables detection of the physical bus error.
With 0 set, the physical bus error is not detected.
With 1 set, the physical bus error is detected.
When this bit is set to 1, the detection result is indicated in the PBER flag in the RLN21nmLiEST register.
For details of the physical bus error, refer to **Section 15.14, Error Status**.

BERE Bit (Bit Error Detection Enable)

The BERE bit enables or disables detection of the bit error.
With 0 set, the bit error is not detected.
With 1 set, the bit error is detected.
When this bit is set to 1, the detection result is indicated in the BER flag in the RLN21nmLiEST register.
For details of the bit error, refer to **Section 15.14, Error Status**.

15.3.3.7 RLN21nmLiCUC — LIN Control Register

Access: This register can be read/written in 8-bit units.

Address: RLN21nmLiCUC: <RLIN21n_base> + $0E_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	OM1	OM0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 15.21 RLN21nmLiCUC Register Contents

Bit Position	Bit Name	Function
7 to 2	Reserved	The write value should be the value after reset.
1	OM1	LIN Mode Select 0: LIN wake-up mode 1: LIN operation mode
0	OM0	LIN Reset 0: LIN reset mode 1: Release from LIN reset mode

Set the RLN21nmLiCUC register to 01_H to cause a transition to LIN wake-up mode after release from LIN reset mode, and set the register to 03_H to cause a transition to LIN operation mode.

In LIN self-test mode, set the RLN21nmLiCUC register to 03_H after a transition to LIN self-test mode is completed.

After a value is written to this register, confirm that the value written is actually indicated in the RLN21nmLiMST register before writing another value.

OM1 Bit (LIN Mode Select)

The OM1 bit selects the specific LIN operation mode (either LIN wake-up mode or LIN operation mode) after release from LIN reset mode.
Setting this bit to 0 causes a transition to LIN wake-up mode.
Setting this bit to 1 causes a transition to LIN operation mode.
This bit is valid only when the OMM0 bit in the RLN21nmLiMST register is 1.
Writing a value to this bit is disabled while the FTS bit in the RLN21nmLiTRC register is 1.

OM0 Bit (LIN Reset)

The OM0 bit selects either a transition to LIN reset mode or release from LIN reset mode.
Setting this bit to 0 causes a transition to LIN reset mode.
Setting this bit to 1 leads to release from LIN reset mode.

15.3.3.8 RLN21nmLiTRC — LIN Transmission Control Register

Access: This register can be read/written in 8-bit units.

Address: RLN21nmLiTRC: <RLIN21n_base> + 10_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	RTS	FTS
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 15.22 RLN21nmLiTRC Register Contents

Bit Position	Bit Name	Function
7 to 2	Reserved	The write value should be the value after reset.
1	RTS	Response Transmission Start 0: Response transmission is stopped in frame separate mode. 1: Response transmission is started in frame separate mode.
0	FTS	Frame Transmission/Wake-up Transmission/Reception Start 0: Frame Transmission/wake-up transmission/reception is stopped. 1: Frame Transmission/wake-up transmission/reception is started.

RTS Bit (Response Transmission Start)

Set the RTS bit to 1 in frame separate mode after header transmission is started (FTS bit is 1) and response transmission data is ready. Once set, this bit is automatically cleared to 0 upon completion of frame communication or transition to LIN reset mode.
Only 1 can be written to this bit; 0 cannot be written.
To write 1 to this bit, write 02_H to the RLN21nmLiTRC register using the store instruction.
Writing a value to this bit is disabled when the OMM0 bit is 0 (in LIN reset mode).
Do not write 1 to this bit when the OMM0 bit is 0 (in LIN wakeup mode).
Writing a value to this bit is disabled when the FTS bit is 0 (frame transmission or wake-up transmission/reception is halted).

FTS Bit (Frame Transmission/Wake-up Transmission/Reception Start)

Set the FTS bit to 1 to start frame/wake-up transmission.
Also set this bit to 1 to allow wake-up reception (counting of the low level width of the input signal).
Only 1 can be written to this bit; 0 cannot be written.
Writing a value to this bit is disabled when the OMM0 bit is 0 (in LIN reset mode). This bit is set to 0 upon completion of frame or wake-up communication and transition to LIN reset mode.

15.3.3.9 RLN21nmLiMST — LIN Mode Status Register

Access: This register can be read/written in 8-bit units.

Address: RLN21nmLiMST: <RLIN21n_base> + 11_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	OMM1	OMM0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 15.23 RLN21nmLiMST Register Contents

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read.
1	OMM1	LIN Mode Status Monitor 0: The LIN master interface is in LIN wake-up mode. 1: The LIN master interface is in LIN operation mode.
0	OMM0	LIN Reset Status Monitor 0: The LIN master interface is in LIN reset mode. 1: The LIN master interface is not in LIN reset mode.

OMM1 Bit (LIN Mode Status Monitor)

The OMM1 bit indicate the current operating mode. The OMM0 bit value is invalid while this bit is 0_B (in LIN reset mode).

OMM0 Bit (LIN Reset Status Monitor)

The OMM0 bit indicates the current operating mode.

15.3.3.10 RLN21nmLiST — LIN Status Register

Access: This register can be read/written in 8-bit units.

Address: RLN21nmLiST: <RLIN21n_base> + 12_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	HTRC	D1RC	—	—	ERR	—	FRC	FTC
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R	R	R	R/W	R/W

Table 15.24 RLN21nmLiST Register Contents

Bit Position	Bit Name	Function
7	HTRC	Successful Header Transmission Flag 0: Header transmission has not been completed. 1: Header transmission has been completed.
6	D1RC	Successful Data 1 Reception Flag 0: Data 1 reception has not been completed. 1: Data 1 reception has been completed.
5, 4	Reserved	The write value should be the value after reset.
3	ERR	Error Detection Flag 0: No error has been detected. 1: Error has been detected.
2	Reserved	The write value should be the value after reset.
1	FRC	Successful Frame/Wake-up Reception Flag 0: Frame or wake-up reception has not been completed. 1: Frame or wake-up reception has been completed.
0	FTC	Successful Frame/Wake-up Transmission Flag 0: Frame or wake-up transmission has not been completed. 1: Frame or wake-up transmission has been completed.

The RLN21nmLiST register is automatically cleared to 00_H upon transition to LIN reset mode and start of the next communication (the FTS bit in the RLN21nmLiTRC register is 1).

In LIN reset mode, writing to the register is disabled.

In LIN reset mode, the register retains 00_H.

Writing to this register is prohibited while the FTS bit in the RLN21nmLiTRC register is 1 (frame transmit or wake-up transmission/reception is started)

To clear the specific bits in the register, write 0 to the bits to be cleared and write 1 to the other bits using the store instruction.

HTRC Flag (Successful Header Transmission Flag)

Only 0 can be written to the HTRC flag; when 1 is written, the bit retains the value that has been retained before 1 is written.
The HTRC flag is set to 1 upon completion of header transmission, but an interrupt request is not generated. To clear the bit to 0 before the next communication (the FTS bit in the RLN21nmLiTRC register is 1), write 0 to the bit in LIN operation mode.

D1RC Flag (Successful Data 1 Reception Flag)

Only 0 can be written to the D1RC flag; when 1 is written, the bit retains the value that has been retained before 1 is written.
The D1RC flag is set to 1 upon completion of data 1 reception. Here, an interrupt request is not generated. To clear the bit to 0 before the next communication, write 0 to the bit in LIN operation mode.

ERR Flag (Error Detection Flag)

The ERR flag is set to 1 upon detection of an error (at least one of the RLN21nmLiEST flags is 1). Here, an interrupt request is generated if the ERRIE bit in the RLN21nmLiIE register is 1 (interrupt request is enabled). To clear the bit to 0 before the next communication (the FTS bit in the RLN21nmLiTRC register is 1), write 0 to the CSER, FER, FTER, PBER, and BER flags in the RLN21nmLiEST register in LIN operation mode or LIN wake-up mode. This clears the ERR flag to 0.

FRC Flag (Successful Frame/Wake-up Reception Flag)

Only 0 can be written to the FRC flag; when 1 is written, the bit retains the value that has been retained before 1 is written.
The FRC flag is set to 1 upon completion of frame or wake-up reception. Here, an interrupt request is generated if the FRCIE bit in the RLN21nmLiIE register is 1 (interrupt request is enabled). To clear the bit to 0 before the next communication (the FTS bit in the RLN21nmLiTRC register is 1), write 0 to the bit in LIN operation mode or LIN wake-up mode.

FTC Flag (Successful Frame/Wake-up Transmission Flag)

Only 0 can be written to the FTC flag; when 1 is written, the bit retains the value that has been retained before 1 is written.
The FTC flag is set to 1 upon completion of frame or wake-up transmission. Here, an interrupt request is generated if the FTCIE bit in the RLN21nmLiIE register is 1 (interrupt request is enabled). To clear the bit to 0 before the next communication(the FTS bit in the RLN21nmLiTRC register is 1), write 0 to the bit in LIN operation mode or LIN wake-up mode.

15.3.3.11 RLN21nmLiEST — LIN Error Status Register

Access: This register can be read/written in 8-bit units.

Address: RLN21nmLiEST: <RLIN21n_base> + 13_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	CSER	—	FER	FTER	PBER	BER
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R	R/W	R/W	R/W	R/W

Table 15.25 RLN21nmLiEST Register Contents

Bit Position	Bit Name	Function
7, 6	Reserved	The write value should be the value after reset.
5	CSER	Checksum Error Flag 0: Checksum error has not been detected. 1: Checksum error has been detected.
4	Reserved	The write value should be the value after reset.
3	FER	Framing Error Flag 0: Framing error has not been detected. 1: Framing error has been detected.
2	FTER	Frame Timeout Error Flag 0: Frame timeout error has not been detected. 1: Frame timeout error has been detected.
1	PBER	Physical Bus Error Flag 0: Physical bus error has not been detected. 1: Physical bus error has been detected.
0	BER	Bit Error Flag 0: Bit error has not been detected. 1: Bit error has been detected.

The RLN21nmLiEST register is automatically cleared to 00_H upon transition to LIN reset mode and start of the next communication (the FTS bit in the RLN21nmLiTRC register is 1).

In LIN reset mode, writing to the register is disabled and the register retains 00_H.

When the FTS bit in the RLN21nmLiTRC register is 1 (frame transmission or wake-up transmission/ reception is started), do not write a value to this register.

To clear the specific bits in the register, write 0 to the bits to be cleared and write 1 to the other bits using the store instruction.

CSER Flag (Checksum Error Flag)

Only 0 can be written to the CSER flag; when 1 is written, the bit retains the value that has been retained before 1 is written.
The CSER flag is set to 1 upon checksum error detection. To clear the bit to 0 before the next communication (the FTS bit in the RLN21nmLiTRC register is 1), write 0 to the bit in LIN operation mode.

FER Flag (Framing Error Flag)

When the FERE bit in the RLN21nmLiEST register is 1 (flaming error detection is enabled) and a flaming error is detected, 1 is written. To clear the bit to 0 before the next communication (the FTS bit in the RLN21nmLiTRC register is 1), write 0 to the bit in LIN operation mode.

FTER Flag (Frame Fimeout Error Flag)

Only 0 can be written to the FTER flag; when 1 is written, the bit retains the value that has been retained before 1 is written.
The FTER flag is set to 1 when the FTERE bit in the RLN21nmLiEDE register is 1 (frame timeout error detection enable) and the frame timeout error is detected. To clear the bit to 0 before the next communication (the FTS bit in the RLN21nmLiTRC register is 1), write 0 to the bit in LIN operation mode.

PBER Flag (Physical Bus Error Flag)

Only 0 can be written to the PBER flag; when 1 is written, the bit retains the value that has been retained before 1 is written.
The PBER flag is set to 1 when the PBERE bit in the RLN21nmLiEDE register is 1 (physical bus error detection enable) and the physical bus error is detected. To clear the bit to 0 before the next communication (the FTS bit in the RLN21nmLiTRC register is 1), write 0 to the bit in LIN operation mode or LIN wake-up mode.

BER Flag (Bit Error Flag)

Only 0 can be written to the BER flag; when 1 is written, the bit retains the value that has been retained before 1 is written.
The BER flag is set to 1 when the BERE bit in the RLN21nmLiEDE register is 1 (bit error detection enable) and the bit error is detected. To clear the bit to 0 before the next communication (the FTS bit in the RLN21nmLiTRC register is 1), write 0 to the bit in LIN operation mode or LIN wake-up mode.

15.3.3.12 RLN21nmLiDFC — LIN Data Field Configuration Register

Access: This register can be read/written in 8-bit units.

Address: RLN21nmLiDFC: <RLIN21n_base> + 14_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	FSM	CSM	RFT	RFDL[3:0]			
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 15.26 RLN21nmLiDFC Register Contents

Bit Position	Bit Name	Function
7	Reserved	The write value should be the value after reset.
6	FSM	Frame Separate Mode Select 0: Frame separate mode is not set. 1: Frame separate mode is set.
5	CSM	Checksum Select 0: Classic checksum mode 1: Enhanced checksum mode
4	RFT	Response Field Communication Direction Select 0: Reception 1: Transmission
3 to 0	RFDL[3:0]	Response Field Length Select b3 b0 0 0 0 0: 0 bytes (+ checksum) 0 0 0 1: 1 byte (+ checksum) 0 0 1 0: 2 bytes (+ checksum) : 0 1 1 1: 7 bytes (+ checksum) 1 0 0 0: 8 bytes (+ checksum) Settings other than the above are prohibited.

Set the RLN21nmLiDFC register when the FTS bit in the RLN21nmLiTRC register is 0 (frame transmission or wake-up transmission/reception is halted).

FSM Bit (Frame Separate Mode Select)

The FSM bit sets the response transmission mode.

With 0 set, frame separate mode is not selected. In this case, after header transmission is started (the FTS bit in the RLN21nmLiTRC register is 1), response is transmitted/received without the RTS bit in the RLN21nmLiTRC register being set.

With 1 set, frame separate mode is selected. When the RTS bit in the RLN21nmLiTRC register is set to 1 during header transmission, response is transmitted after header transmission is completed.

Header transmission is successfully completed (the HTRC bit in the RLN21nmLiST register is 1).

Response transmission is requested (the RTS bit in the RLN21nmLiTRC register is 1).

For response reception (the RFT bit is 0), set the FSM bit to 0.
When causing a transition to LIN self-test mode, set this bit to 0 before transition.
For details of frame separate mode, refer to **Section 15.11.1, Transmission of LIN Frames**.

CSM Bit (Checksum Select)

The CSM bit sets the checksum mode.
With 0 set, classic checksum mode is selected.
With 1 set, enhanced checksum mode is selected.
When the frame timeout error is used (the FTERE bit in the RLN21nmLiEDE register is 1), the specific timeout time depends on the setting of this bit. For details of the bit error, refer to **Section 15.14, Error Status**.

RFT Bit (Response Field Communication Direction Select)

The RFT bits set the direction of the response field communication and wake-up signal.
With 0 set, reception is performed in the response field. In LIN wake-up mode, wake-up reception is performed (low level width of the input signal is counted).
With 1 set, transmission is performed in the response field. In LIN wake-up mode, wake-up transmission is performed.

RFDL[3:0] Bits (Response Field Length Select)

The RFDL bits set the length of the response field data.
The data length can be 0 to 8 bytes excluding the checksum size.

15.3.3.13 RLN21nmLiIDB — LIN ID Buffer Register

Access: This register can be read/written in 8-bit units.

Address: RLN21nmLiIDB: <RLIN21n_base> + 15_H

Value after reset: Undefined

Bit	7	6	5	4	3	2	1	0
	IDP1	IDP0	ID[5:0]					
Value after reset	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 15.27 RLN21nmLiIDB Register Contents

Bit Position	Bit Name	Function
7	IDP1	Parity Setting (P1) Sets the parity bits (P1) to be transmitted in the ID field.
6	IDP0	Parity Setting (P0) Sets the parity bits (P0) to be transmitted in the ID field.
5 to 0	ID[5:0]	ID Setting Sets the 6-bit ID value to be transmitted in the ID field.

Set the RLN21nmLiIDB register when the FTS bit in the RLN21nmLiTRC register is 0 (frame transmission or wake-up transmission/reception is halted).

In LIN self-test mode, this register operates as follows:

Write the value to be transmitted to the register before communication. After flame transmission is completed (after loopback), the inverse of the value transmitted can be read from the register.

For details of LIN self-test mode, see **Section 15.15, LIN Self-Test Mode**.

IDP Bits (Parity Setting)

The IDP bits set the parity bits (P0 and P1) to be transmitted in the ID field of the LIN frame. IDP0 is P0 and IDP1 is P1. Since parity is not automatically calculated, set the calculation result. Note that if the erroneous result is set, it is transmitted as is.

ID Bits (ID Setting)

The ID bit sets the 6-bit ID value to be transmitted in the ID field of the LIN frame.

15.3.3.14 RLN21nmLiCBR — LIN Checksum Buffer Register

Access: This register can be read/written in 8-bit units.
However, this register can be read/written in 8-bit units in LIN self-test mode.

Address: RLN21nmLiCBR: <RLIN21n_base> + 16_H

Value after reset: Undefined

Bit	7	6	5	4	3	2	1	0
	CKSM[7:0]							
Value after reset	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 15.28 RLN21nmLiCBR Register Contents

Bit Position	Bit Name	Function
7 to 0	CKSM[7:0]	Holds the checksum value transmitted or received.

In LIN operation mode, this register operates as follows:

- When the RFT bit in the RLN21nmLiDFC register is 1 (transmission):
  The value transmitted can be read from the register. Read the value after transmission is completed.
  Writing to this register is invalid.
- When the RFT bit in the RLN21nmLiDFC register is 0 (reception):
  The value received can be read from the register. Read the value after reception is completed.
  Writing to this register is invalid.

In LIN self-test mode, this register operates as follows:

- When the RFT bit in the RLN21nmLiDFC register is 1 (transmission):
  The inverse of the value received can be read from the register after frame transmission/reception completion (after loop-back).
  Writing to this register is invalid.
- When the RFT bit in the RLN21nmLiDFC register is 0 (reception):
  The value to be received should be written to the register before communication.
  The inverse of the value received can be read from the register after frame transmission/reception is completed (after loop-back).

See **Section 15.15, LIN Self-Test Mode** for the details on LIN self-test mode.

Set the RLN21nmLiCBR register when the FTS bit in the RLN21nmLiTRC register is 0 (frame transmission or wake-up transmission/reception is halted).

15.3.3.15 RLN21nmLiDBRb — LIN Data Buffer b Register

Access: This register can be read/written in 8-bit units.

Address: RLN21nmLiDBR1: <RLIN21n_base> + 18_H, RLN21nmLiDBR2: <RLIN21n_base> + 19_H, RLN21nmLiDBR3: <RLIN21n_base> + $1A_H$, RLN21nmLiDBR4: <RLIN21n_base> + $1B_H$, RLN21nmLiDBR5: <RLIN21n_base> + $1C_H$, RLN21nmLiDBR6: <RLIN21n_base> + $1D_H$, RLN21nmLiDBR7: <RLIN21n_base> + $1E_H$, RLN21nmLiDBR8: <RLIN21n_base> + $1F_H$

Value after reset: Undefined

Bit	7	6	5	4	3	2	1	0
	LDB[7:0]							
Value after reset	—	—	—	—	—	—	—	—
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 15.29 RLN21nmLiDBRm (m = 1 to 8) Register Contents

Bit Position	Bit Name	Function
7 to 0	LDB[7:0]	Sets the data to be transmitted or allows the received data to be read. Setting Range: 00_H to FF_H

- For response transmission:

  The RLN21nmLiDBRm registers set the data to be transmitted in the response field.
  Use these registers with the following settings.

  – RFT in RLN21nmLiDFC register is 1 (transmission)
  – FSM in RLN21nmLiDFC register is 0 (not frame separate mode)
  – FTS bit in RLN21nmLiTRC register is 0 (frame transmission or wake-up transmission/ reception is halted)

  or

  – RFT in RLN21nmLiDFC register is 1 (transmission)
  – FSM in RLN21nmLiDFC register is 1 (frame separate mode)
  – RTS in RLN21nmLiTRC register is 0 (response transmission is halted)

- For response reception:

  The RLN21nmLiDBRm registers hold the data received in the response field.
  The received data is overwritten. If an error is detected, the data prior to reception interruption is stored in the register.

  Do not read these registers when the FTS bit is 1 (frame transmission or wake-up transmission/ reception is started)

In LIN self-test mode, these registers operate as follows:

The value to be transmitted should be written to the registers before communication.

The inverse of the value received can be read from the register after frame transmission/reception completion (after loop back).

See **Section 15.15, LIN Self-Test Mode** for the details on the LIN self-test mode.

15.4 Interrupt Sources

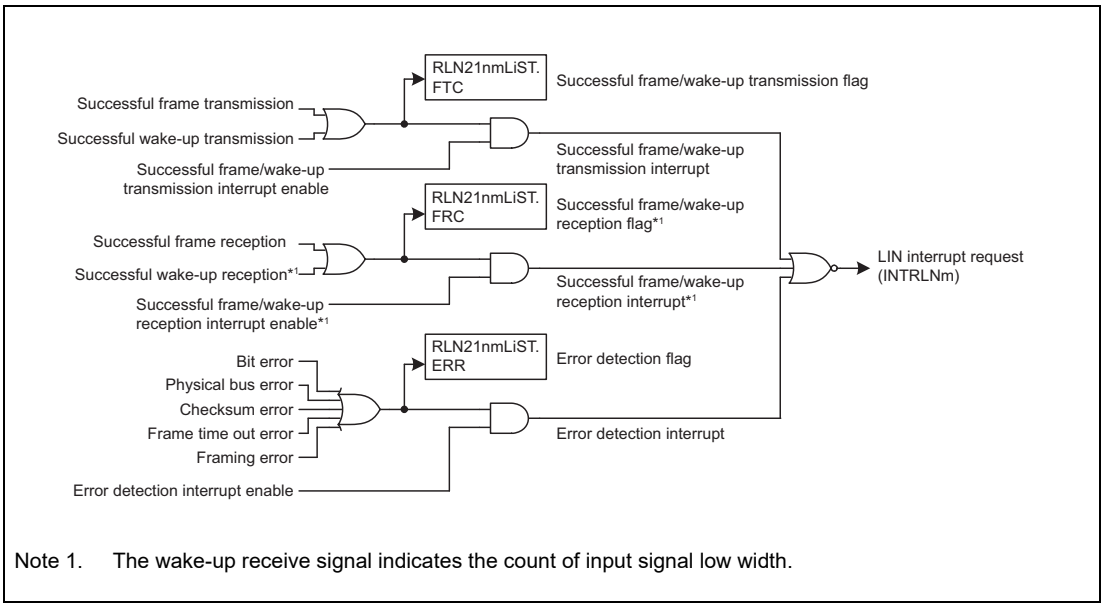
The LIN interrupts are interrupt requests generated by the LIN master interface.

There are three interrupt factors for each channel; frame/wake-up transmission completion, frame/wake-up reception completion, and error detection.

The interrupt request from three interrupt states, frame/wake-up transmit completion, frame/wake-up reception completion, and error detection, is ORed to be one interrupt request “LIN interrupt”.

The respective interrupt request is output when the corresponding flag in the RLN21nmLiST register is set to 1 while the corresponding bit in the RLN21nmLiIE register is 1 (interrupt enabled). However, if an interrupt is requested when the corresponding flag in the RLN21nmLiST register has been set to 1, it is ignored. Therefore, clear the corresponding flag to 0 to enable the interrupt.

Figure 15.2 shows a block diagram of the LIN interrupt.



Note 1. The wake-up receive signal indicates the count of input signal low width.

Figure 15.2 LIN Interrupt Block Diagram

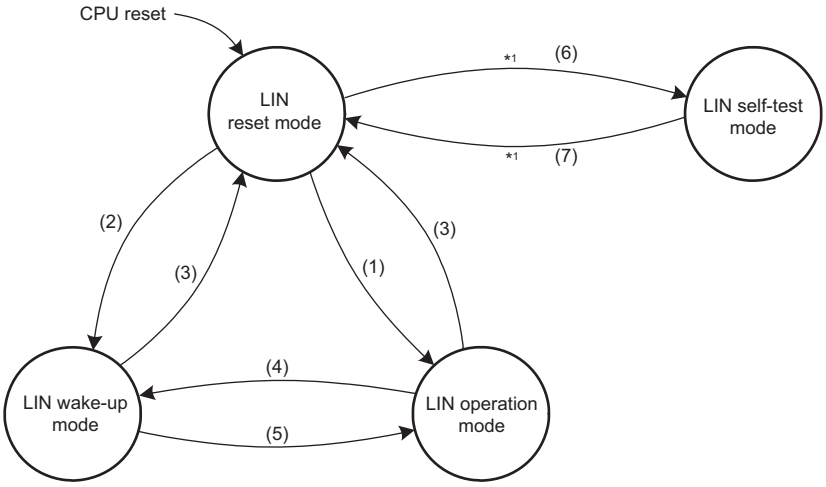
15.5 Modes

The LIN Master Interface provides the following four modes, depending upon the specific function to be performed:

- LIN reset mode
- LIN operation mode
- LIN wake-up mode
- LIN self-test mode

The transition of the operating modes except LIN self-test mode is controlled independently for respective channels.

Figure 15.3 shows mode transitions. **Table 15.30** describes mode transition conditions. **Table 15.31** lists operations available in each mode.



Note 1. See **Section 15.15, LIN Self-Test Mode** for the details.

Figure 15.3 Mode Transitions

Table 15.30 Transition Condition of Each Mode

	Mode transition			Transition condition
1	LIN reset mode	→	LIN operation mode	RLN21nmLiCUC.OM1, OM0 = 11_B
2	LIN reset mode	→	LIN wake-up mode	RLN21nmLiCUC.OM1, OM0 = 01_B
3	LIN wake-up mode LIN operation mode	→	LIN reset mode	RLN21nmLiCUC.OM0 = 0_B
4	LIN operation mode	→	LIN wake-up mode	RLN21nmLiCUC.OM1, OM0 = 01_B
5	LIN wake-up mode	→	LIN operation mode	RLN21nmLiCUC.OM1, OM0 = 11_B
6	LIN reset mode	→	LIN self-test mode	See **Section 15.15, LIN Self-Test Mode**.
7	LIN self-test mode	→	LIN reset mode	See **Section 15.15, LIN Self-Test Mode**.

Table 15.31 Operations Available in Each Mode

LIN operation mode	LIN wake-up mode	LIN self-test mode
Header transmission Response transmission Response reception Error detection	Wake-up transmission Wake-up reception Error detection	Self test

Whether a transition has been made to LIN reset mode, the LIN mode, or the UART mode can be verified by reading the OMM1 and OMM0 bits in the RLN21nmLiMST register.

For a description of the LIN self-test mode, see **Section 15.15, LIN Self-Test Mode**.

15.6 LIN Reset Mode

Setting the OM0 bit in the RLN21nmLiCUC register to 0_B (LIN reset mode) causes a transition to LIN reset mode. The change to LIN reset mode can be verified by determining that the OMM0 bit in the RLN21nmLiMST register has been set to 0_B (LIN reset mode). In this mode, the LIN communication stops.

From LIN reset mode, transitions to LIN operation mode, LIN wake-up mode, and LIN self-test mode can be made.

When the mode changes to LIN reset mode, the following registers are initialized to their reset values, and as long as LIN reset mode is in effect, they retain their value after resets.

- RLN21nmLiTRC register
- RLN21nmLiST register
- RLN21nmLiEST register

The following registers retain their previous values even when a transition to LIN reset mode is made:

- RLN21nGLWBR register
- RLN21nGLBRP0 register
- RLN21nGLBRP1 register
- RLN21nmLiMD register
- RLN21nmLiBFC register
- RLN21nmLiSC register
- RLN21nmLiWUP register
- RLN21nmLiIE register
- RLN21nmLiEDE register
- RLN21nmLiDFC register
- RLN21nmLiIDB register
- RLN21nmLiCBR register
- RLN21nmLiDBRb register

15.7 LIN Operation Mode

In LIN operation mode, frame processing (header transmission, header reception, response transmission, response reception, and error detection) can be performed.

During a transition from LIN reset mode to LIN mode, setting the OM1 and OM0 bits in the RLN21nmLiCUC register to 11_B changes the mode to LIN operation mode, changing the OMM1 and OMM0 bits in the RLN21nmLiMST register to 11_B. Communication settings should be performed after the RLN21nmLiMST register has become 11_B.

15.8 LIN Wake-up Mode

In LIN wake-up mode, wake-up signal processing (wake-up transmission, wake-up reception, and error detection) can be performed.

During a transition from LIN reset mode to LIN mode, setting the OM1 and OM0 bits in the RLN21nmLiCUC register to 01_B changes the mode to LIN wake-up mode, changing the OMM1 and OMM0 bits in the RLN21nmLiMST register to 01_B. Communication settings should be performed after the RLN21nmLiMST register has become 01_B.

15.9 Header Transmission/Response Transmission/Response Reception

15.9.1 Header Transmission

Figure 15.4 shows the operation of the LIN master interface (LIN master mode) in header transmission. **Table 15.32** provides processing in header transmission.

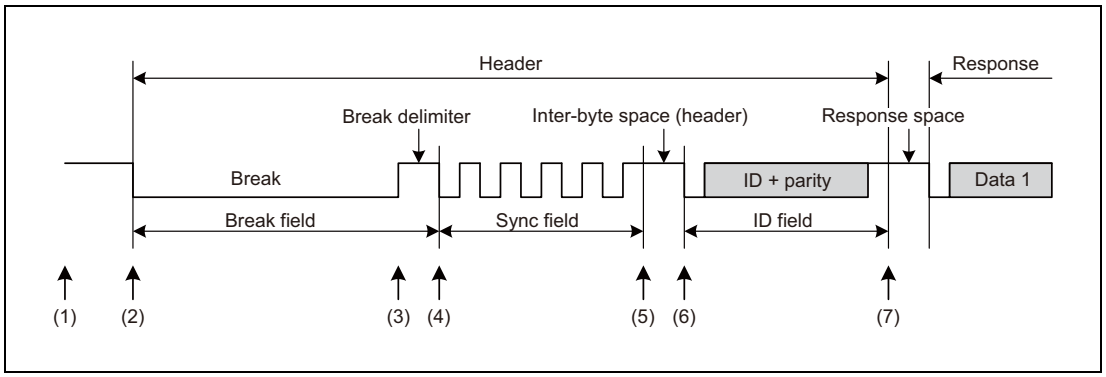


Figure 15.4 Operation in Header Transmission

Table 15.32 Processing in Header Transmission

	Software processing	LIN Master Interface processing
(1)	• Sets a baud rate • Enables interrupt • Enables error detection • Sets frame configuration parameters • Changes LIN operation mode • Sets information on the frame to be transmitted (ID, parity, data length, response direction, Checksum method, and transmission data)	Waits for the setting of the FTS bit in the RLN21nmLiTRC register to 1 by software (idle).
(2)	Sets the FTS bit in the RLN21nmLiTRC register to 1 (frame transmission or wake-up transmission/reception started)	Transmits a break.
(3)	Waits for an interrupt request	Transmits a break delimiter.
(4)		Transmits a sync field (55_H).
(5)		Transmits an inter-byte space (header).
(6)		Transmits an ID field.
(7)		Sets a successful header transmission flag.

NOTE

For information about error detection, refer to **Section 15.14, Error Status**.

15.9.2 Response Transmission

Figure 15.5 shows the operation of the LIN Master Interface (LIN master mode) in response transmission. **Table 15.33** provides processing in response transmission.

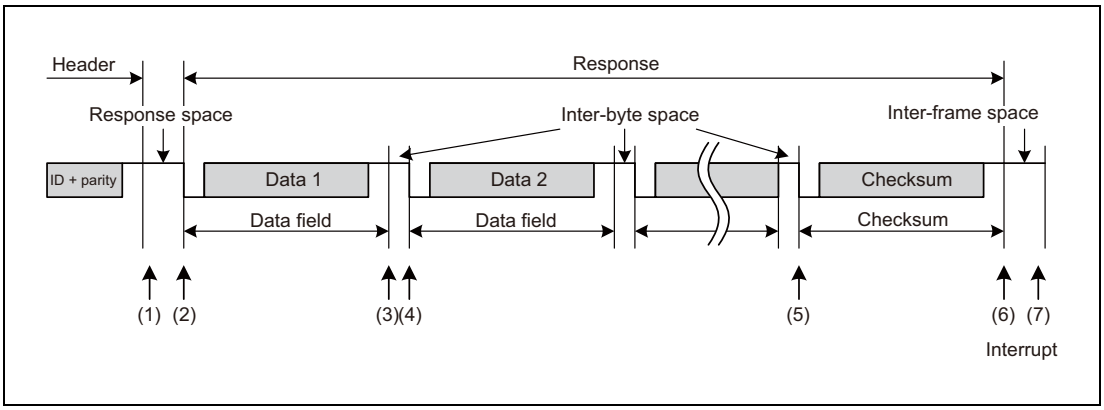


Figure 15.5 Operation in Response Transmission

Table 15.33 Processing in Response Transmission

	Software processing	LIN Master Interface processing
(1)	(When in frame separate mode) • Sets the RTS bit in the RLN21nmLiTRC register to 1 (response transmission started) (When not in frame separate mode) • Waits for an interrupt request	(When in frame separate mode) • Waits for the setting of the RTS bit in the RLN21nmLiTRC register to 1 by software (1 is output during this time). • When the bit is set to 1, sends a response space. (When not in frame separate mode) • Sends a response space.
(2)	Waits for an interrupt request	Transmits the data 1.
(3)		Transmits an inter-byte space.
(4)		• Transmits the data 2. • Transmits an inter-byte space • Transmits the data 3. • Transmits an inter-byte space (Repeats the transmission of inter-byte spaces as many times as the data length specified in bits RFDL[3:0] in the RLN21nmLiDFC register.) : :
(5)		Transmits the checksum.
(6)		• Sets a successful frame/wake-up transmission flag. • Sets the FTS bit in the RLN21nmLiTRC register to 0 (frame transmission or wake-up transmission/reception stopped) (When in frame separate mode). • Sets the RTS bit in the RLN21nmLiTRC register to 0 (response transmission stopped).
(7)	• Processing after communication Checks the RLN21nmLiST register, and clears flags.	Idle

NOTE

For information about error detection, refer to **Section 15.14, Error Status**.

15.9.3 Response Reception

Figure 15.6 shows the operation of the LIN master interface (LIN master mode) on response reception. **Table 15.34** provides processing in response reception.

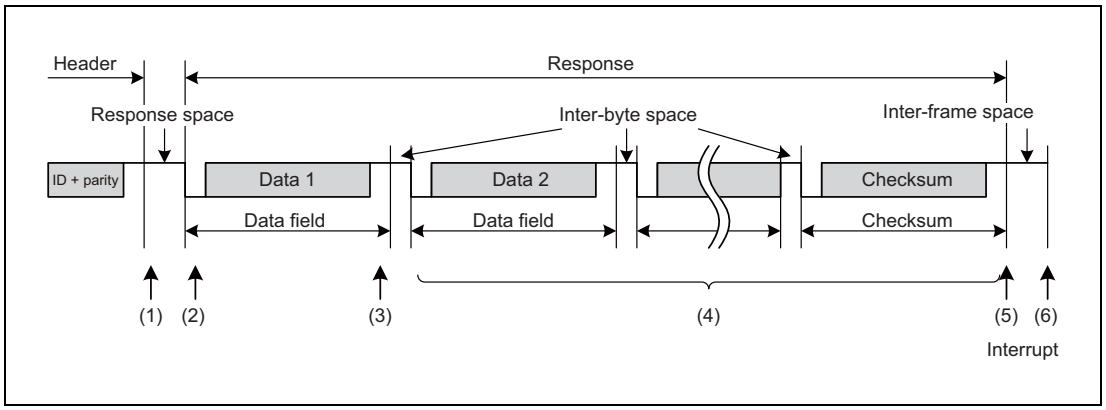


Figure 15.6 Operation in Response Reception

Table 15.34 Processing in Response Reception

	Software processing	LIN Master Interface processing
(1)	Waits for an interrupt request (no processing).	Waits for detection of a start bit.
(2)	Waits for an interrupt request.	Receives the data 1 when the start bit is detected.
(3)		Sets the successful data 1 reception flag.
(4)		• Receives the data 2 when the start bit is detected. • Receives the data 3 when the start bit is detected. Repeats the transmission of inter-byte spaces as many times as the data length specified in bits RFDL[3:0] in the RLN21nmLiDFC register.) ⋮ • Receives the checksum when the start bit is detected.
(5)		• Determines the checksum. • Sets the successful frame/wake-up reception flag. • Sets the FTS bit in the RLN21nmLiTRC register to 0 (frame transmission or wake-up transmission/reception stopped).
(6)	• Processing after communication Reads the received data. Checks the RLN21nmLiST register, and clears flags.	Idle

NOTE

For information about error detection, refer to **Section 15.14, Error Status**.

15.10 Data Transmission/Reception

15.10.1 Data Transmission

One bit of data is transmitted per 1 Tbit.

The data that is transmitted returns to the reception data input pin via the LIN transceiver. The received data and the transmitted data are compared bit by bit, and the results are stored in the BER flag in the RLN21nmLiEST register (see **Section 15.14, Error Status**).

In LIN Master interface, the sampling point for received data, 1 Tbit is generated to be 16fLIN, and thus is at the 13th clock cycle (81.25% position).

Figure 15.7 shows an example of data transmission timing.

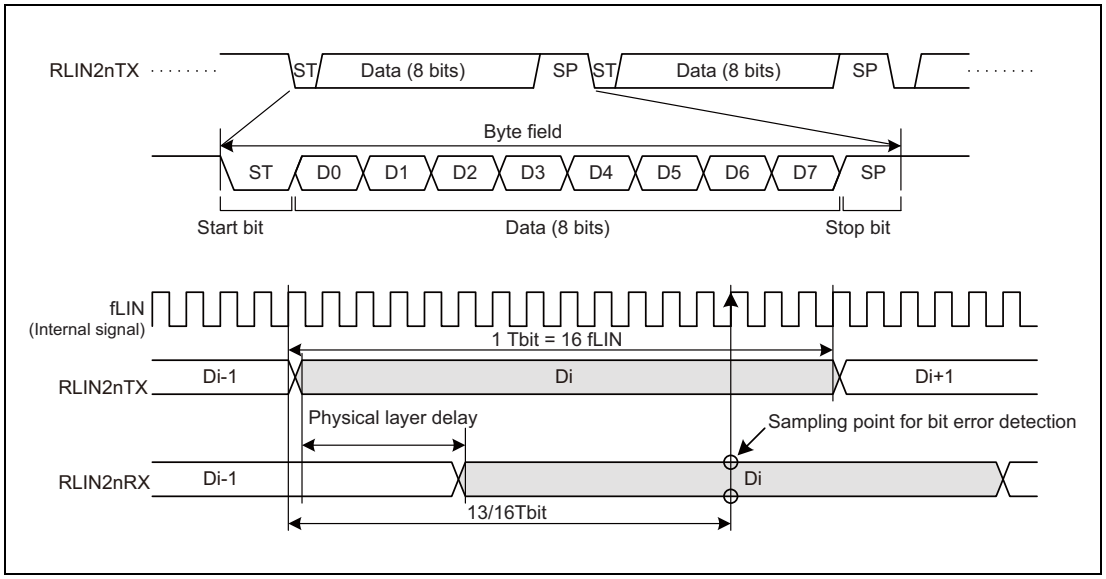


Figure 15.7 Example of Data Transmission Timing

15.10.2 Data Reception

Data reception is performed by using the synchronized RLIN2nRX signal (an internal signal) that is the input from the LRX0 pin synchronized with the LIN system clock (fLIN).

The byte field is synchronized at the falling edge of the start bit for the synchronized RLIN2nRX signal. After the falling edge is detected, sampling is performed again 0.5 Tbit later, and the falling edge is recognized as a start bit if the synchronized RLIN2nRX signal is low level. The falling edge is not recognized as a start bit if the RLIN2nRX signal after the clearing of the resetting is low-level-fixed or if a high level is detected on re-sampling.

After the start bit is detected, the system samples 1 bit per Tbit.

Figure 15.8 shows an example of data reception timing.

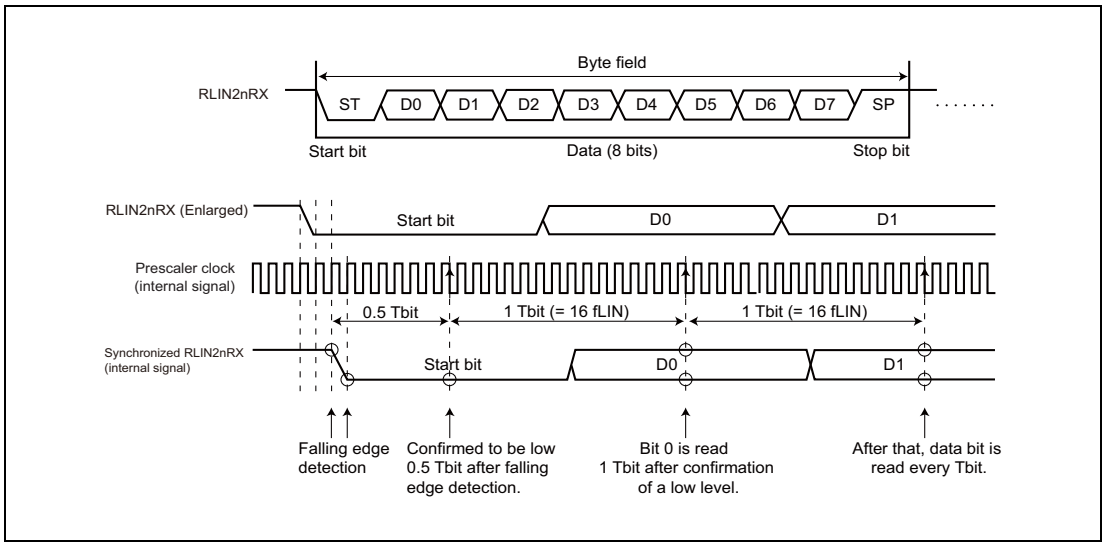


Figure 15.8 Example of Data Reception Timing

15.11 Transmit/Receive Data Buffering

This section explains the buffer processing that takes place when the LIN Master Interface sends or receives data continuously.

15.11.1 Transmission of LIN Frames

For an 8-byte transmission, the contents stored in registers RLN21nmLiDBR1 to RLN21nmLiDBR8 are sequentially transmitted to data areas 1 to 8 of the LIN frame. In the case of a 4-bytes transmission, the contents stored in registers RLN21nmLiDBR1 to RLN21nmLiDBR4 are transmitted to data areas 1 to 4 of the LIN frame, but the contents of registers RLN21nmLiDBR5 to RLN21nmLiDBR8 are not transmitted. The transmitted checksum data is stored in the RLN21nmLiCBR register.

Figure 15.9 depicts the LIN transmission processing and the required buffer.

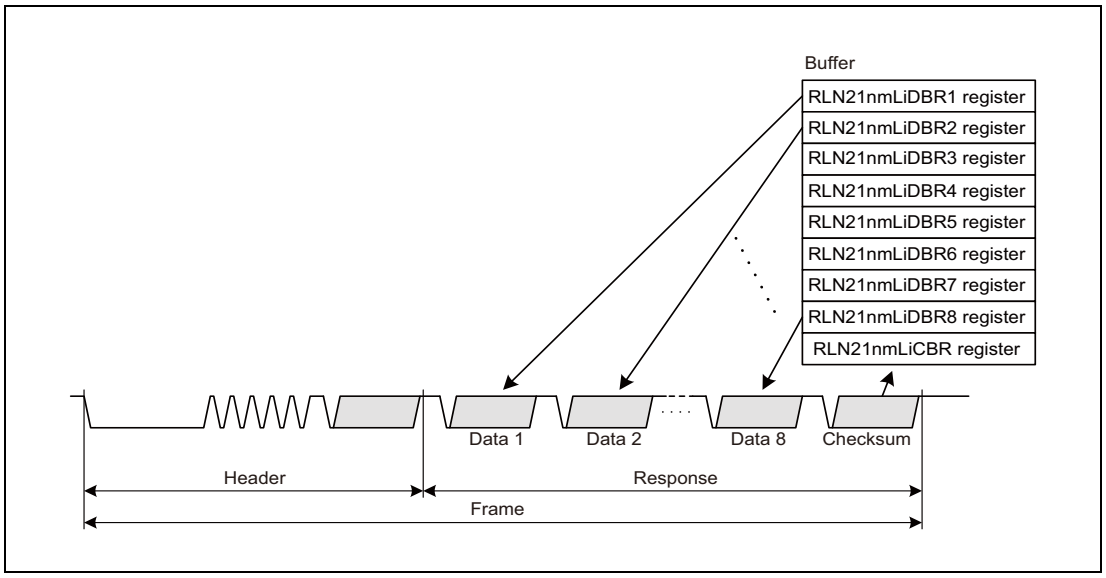


Figure 15.9 LIN Transmission Processing and Required Buffer

(1) Frame Separate Mode

Setting the FSM bit in the RLN21nmLiDFC register to 1 turns on the frame separate mode.

In frame separate mode, a header and a response are transmitted when prompted by separate transmission start requests.

When the transmission of a header is finished, the HTRC flag in the RLN21nmLiST register turns 1 (successful header transmission).

15.11.2 Reception of LIN Frames

For an 8-byte reception, the contents of data areas 1 to 8 of the LIN frame is stored in registers RLN21nmLiDBR1 to RLN21nmLDBR8, respectively, upon receipt of a stop bit. In the case of a 4-byte reception, the contents of data areas 1 to 4 of the LIN frame are stored in registers RLN21nmLiDBR1 to RLN21nmLiDBR4, respectively; however, no data is stored in registers RLN21nmLiDBR5 to RLN21nmLiDBR8. Also, the received checksum data is stored in the RLN21nmLiCBR register.

Figure 15.10 depicts the LIN reception processing and the required buffer.

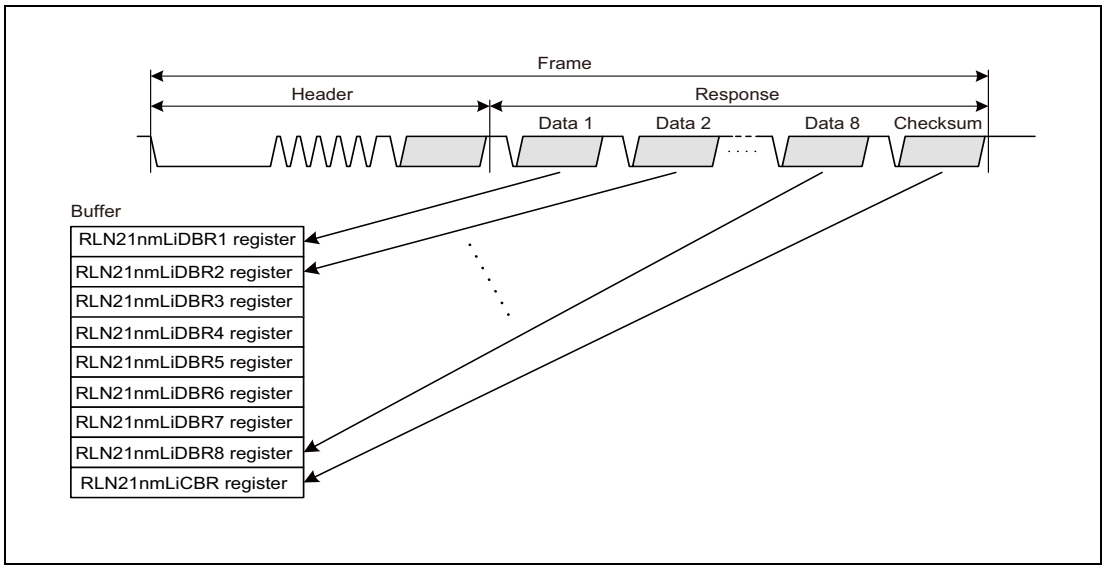


Figure 15.10 LIN Reception Processing and Required Buffer

(1) Reception of Data 1

When the reception of the first byte of data is finished, the D1RC flag in the RLN21nmLiST register turns 1 (successful data 1 reception).

15.12 Wake-up Transmission/Reception

The wake-up transmission/reception can be used in LIN wake-up mode.

15.12.1 Wake-up Transmission

In LIN wake-up mode, setting the RFT bit in the RLN21nmLiDFC register to 1 (transmission) and the FTS bit in the RLN21nmLiTRC register to 1 (frame transmission or wake-up transmission/reception started) causes a wake-up signal to be output from the output pin. The low level width of the wake-up signal should be set using the WUTL[3:0] bits in the RLN21nmLiWUP register.

However, when the setting of the LWBR0 bit in the RLN21nGLWBR register is 1 (if LIN2.x is in use), the width at low level of the LIN system clock (fLIN) becomes that of fa regardless of the setting of LCKS bits in the RLN21nmLiMD register. Setting the bit rate when 19200 bps is selected as fa and setting the WUTL[3:0] bits in the RLN21nmLiWUP register to 0100_B (5 Tbits) enables output of a signal with width at low level of 260 µs in LIN wakeup mode regardless of the setting of LCKS bits in the RLN21nmLiMD register.

If a wake-up low is output without any error, the FTC flag in the RLN21nmLiST register turns 1 (successful frame or wake-up transmission); when the FTCIE bit in the RLN21nmLiIE register is 1 (successful frame/wakeup transmission interrupt enabled), an interrupt request is generated.

If an error is detected, wake-up transmission is suspended and the error flag of the detected error (PBER flag or BER flag in the RLN21nmLiEST register) is set to 1 (detection of a physical bus error or bit error).

Figure 15.11 shows the wake-up transmission timing.

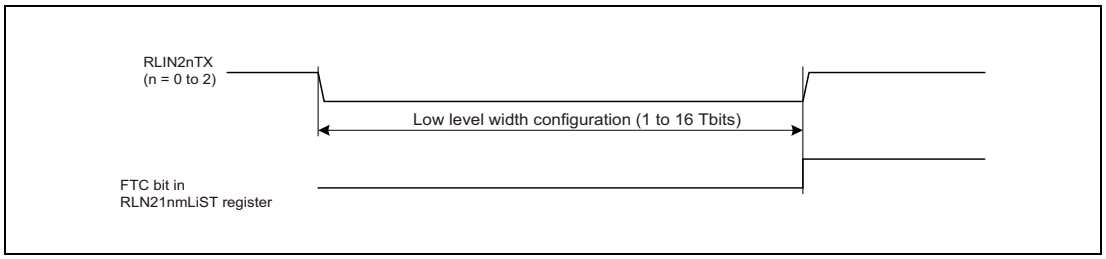


Figure 15.11 Wake-up Transmission Timing

15.12.2 Wake-up Reception

The detection of a wake-up signal involves the use of an input signal low level width count function.

The input signal low level width count function measures the low level width of the input signal to the LRX0 pin, using the same sampling point as data reception. This allows the 2.5-Tbit or longer low-level width of the input signal of fLIN to be measured.

When LIN Specification Package Revision 1.3 is used, set the LWBR0 bit in the RLN21nmGLWBR register to 0. When LIN Specification Package Revision 2.x is used, set the RLN21nmGLWBR register to 1.

When LWBR0 bit is set to 1, regardless of the setting of the LCKS bit in the RLN21nmLiMD register, fa is selected as the LIN system clock (fLIN) (the LCKS bit is not changed).

Setting the baud rate to 19200 bps while fa is selected allows the 130µs or longer low-level width of the input signal to be detected during LIN wake-up mode regardless of the setting of the RLN21nmLiMD register.

When using this function, in LIN wake-up mode set the RFT bit in the RLN21nmLiDFC register to 0 (reception), and the FTS bit in the RLN21nmLiTRC register to 1 (frame transmission or wake-up transmission/reception a started).

When the low level width to be measured is reached, the FRC flag in the RLN21nmLiST register turns 1 (successful frame/wake-up reception). If the FRCIE bit in the RLN21nmLiIE register is 1 (successful frame or wake-up reception interrupt enabled), an interrupt request is generated.

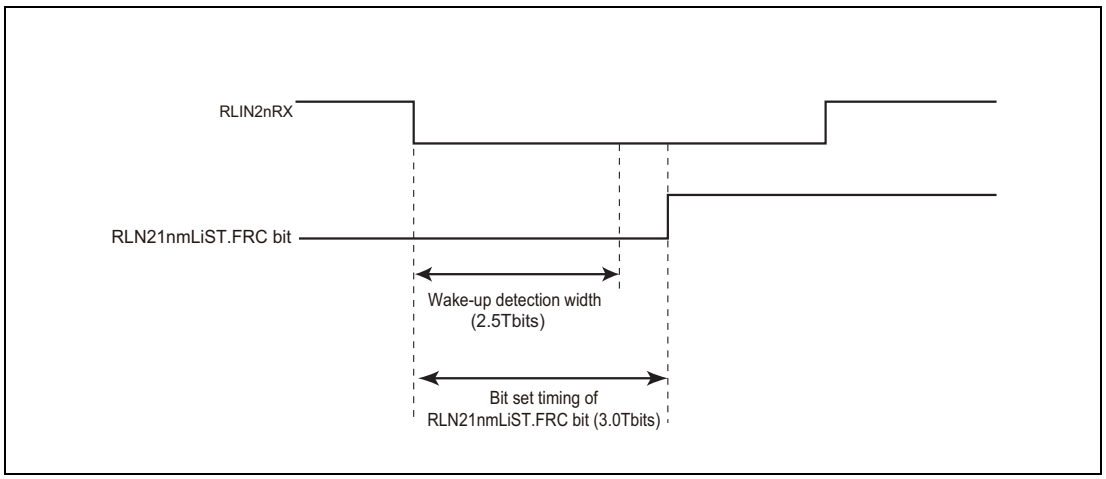


Figure 15.12 Input Signal Low Level Count Function

15.12.3 Wakeup Collision

If the master node and the slave node transmit wakeup signals simultaneously, a collision will occur on the LIN bus, though a collision of wakeup signals is not detected in the LIN master interface.

15.13 Status

During LIN mode operation, the LIN Master Interface can detect seven types of statuses.

The three statuses, successful frame/wake-up transmission, successful frame/wake-up reception, error detection, scan generate interrupt requests.

Table 15.35 shows the types of statuses available.

Table 15.35 Types of Statuses

Status	Status Set Condition	Status Clear Condition	Operation Mode Capable of Status Detection	Corresponding Bit	Interrupt
Reset	After the OM0 bit in the RLN21nmLiCUC register is set to not-LIN–reset-mode, if actually the LIN Master Interface is cleared from LIN reset mode.	After the OM0 bit in the RLN21nmLiCUC register is set to LIN reset mode, if actually the LIN Master Interface enters LIN reset mode.	All modes	OMM0 bit in RLN21nmLiMST register	—
Operation mode	After the OM1 bit in the RLN21nmLiCUC register is set to LIN operation mode, if actually the LIN Master Interface enters LIN operation mode.	After the OM1 bit in the RLN21nmLiCUC register is set to LIN wake-up mode, if actually the LIN Master Interface enters LIN wake-up mode.	• LIN operation mode • LIN wake-up mode	OMM1 bit in RLN21nmLiMST register	—
Frame/wake-up transmission end	When a frame (header transmission + response transmission), a wake-up signal, or a data group is transmitted successfully.	• When another communication is started • When cleared by software • After transition to LIN reset mode	• LIN operation mode • LIN wake-up mode	FTC flag in RLN21nmLiST register	√
Frame/wake-up reception end	When a frame (header transmission + response reception), a wake-up signal, or a data group is received successfully.	• When another communication is started • When cleared by software • After transition to LIN reset mode	• LIN operation mode • LIN wake-up mode	FRC flag in RLN21nmLiST register	√
Error detection	If any of the PRER flag, CSER flag, FER flag, FTER flag, PBER flag, and BER flags in the RLN21nmLiEST register turns 1 (error detected).	• When another communication is started • When cleared by software *1 • After transition to LIN reset mode	• LIN operation mode • LIN wake-up mode	ERR flag in RLN21nmLiST register	√
Data 1 reception end	The RFT bit in the RLN21nmLiDFC register is 0 (reception) and the first byte of the response field is received.*2	• When another communication is started • When cleared by software • After transition to LIN reset mode	LIN operation mode	D1RC flag in RLN21nmLiST register	—
Header reception end	When a header field is transmitted successfully.	• When another communication is started • When cleared by software • After transition to LIN reset mode	LIN operation mode	HTRC flag in RLN21nmLiST register	—

Note 1. In LIN operation mode, the ERR flag in the RLN21nmLiST register is cleared to 0 by writing 0 to the CSER flag, FER flag, FER flag, FTER flag, PBER flag or BER flags in the RLN21nmLiEST register.

Note 2. Not detected when the RFDL [3:0] bits in the RLN21nmLiDFC register are 0000_B (0-byte + checksum).

15.14 Error Status

15.14.1 Types of Error Status

The LIN Master Interface can detect five types of error status in LIN master mode. The condition of these error statuses can be checked by means of the corresponding bits in the RLN21nmLiEST register.

All error statuses represent interrupt events.

Table 15.36 shows the types of error status.

Table 15.36 Types of Error Status

Status	Error Detection Condition	Operation Mode Capable of Error Detection	Commu-nication	Enable/ Disable Detection	Corresponding Bit
Bit error	The transmitted data and the data on the LIN bus monitored by the receive pin do not match *1	• LIN operation mode • LIN wake-up mode	Suspended	√	BER flag in RLN21nmLiEST register
Physical bus error	• LIN bus is detected to be high level when sending a break • LIN bus is detected to be low level when sending a break delimiter • LIN bus is detected to be high level when sending a wake-up	• LIN operation mode • LIN wake-up mode	Suspended	√	PBER flag in RLN21nmLiEST register
Timeout error	A frame or response transmission/reception does not terminate within a given time*2	LIN operation mode	Suspended	√	FTER flag in RLN21nmLiEST register
Framing error	In response field reception, a stop bit of each data byte is low level	LIN operation mode	Suspended	√	FER flag in RLN21nmLiEST register
Checksum error	In response field reception, the result of checksum test gives an error	LIN operation mode	—	—	CSER flag in RLN21nmLiEST register

Note 1. If a bit error is detected, processing is stopped after a stop bit is sent. If a bit error is detected in a non-data area, such as an inter-byte space, the transmission is suspended immediately after the bit which had the error is sent. If a bit error is detected during the transmission of a wake-up, the transmission of the wake-up is suspended after the bit which had the error is sent.

Note 2. The timeout time depends on the response field data length (the RFDL 3:0] bits in the RLN21nmLiDFC register) and the checksum selection (the CSM bit in the RLN21nmLiDFC register), and this can be calculated according to the following formula:
On classic selection (when the CSM bit in RLN21nmLiDFC is 0): Timeout time = 49 + (number of data bytes + 1) × 14 [Tbit]
On enhanced selection (when the CSM bit in RLN21nmLiDFC is 1): Timeout time = 48 + (number of data bytes + 1) × 14 [Tbit]

The aforementioned timeout time is a time greater than the TFRAME_MAX of LIN Specification Package Revision 1.3 on classic selection, or the TFRAME_MAX of LIN Specification Package Revision 2.x on enhanced selection.

The error status is cleared when the next communication is started, by software, or at a transition to LIN reset mode.

15.14.2 Target Time Area for Error Detection

Figure 15.13 shows the time domain in which the LIN master interface performs monitoring for error detection.

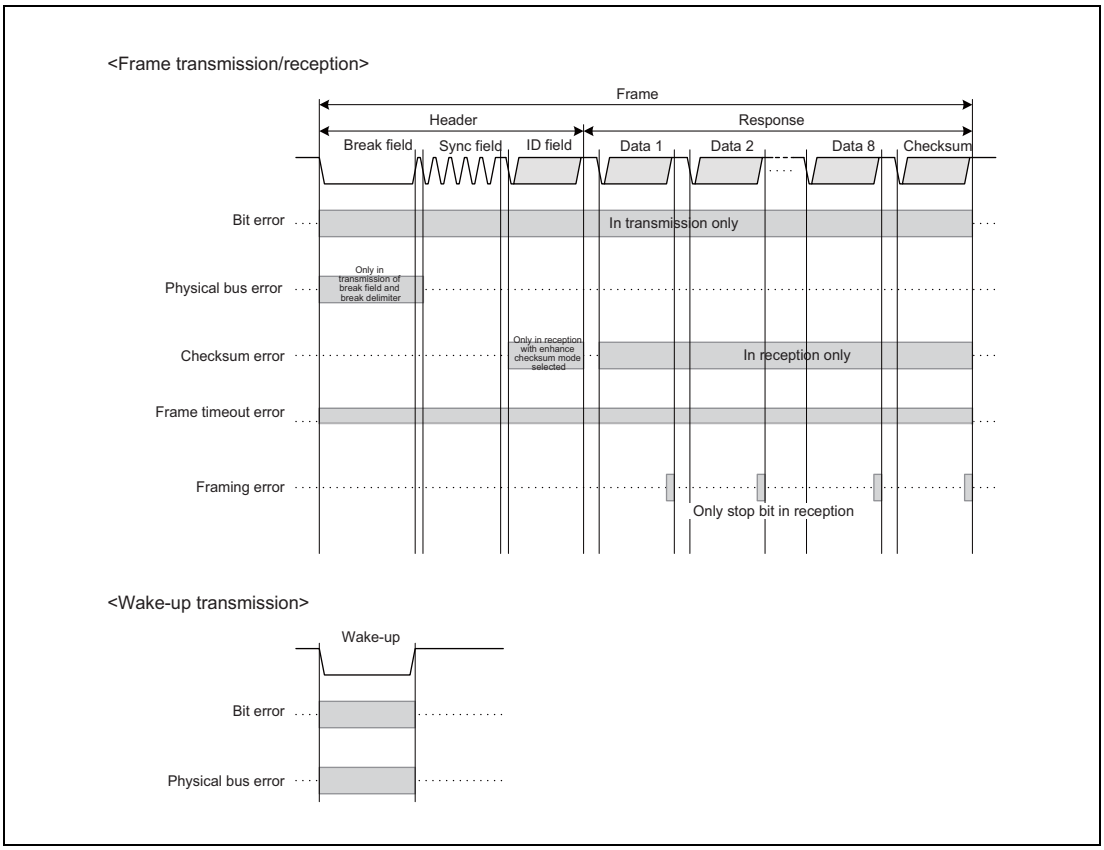


Figure 15.13 Target Time Area for Error Detection

15.15 LIN Self-Test Mode

The LIN Master Interface provides LIN self-test mode. When LIN self-test mode is turned on, RLIN2nTX and RLIN2nRX are disconnected from the external pin, and the internal RLIN2nTX and RLIN2nRX are connected. Thus, the frame transmitted from RLIN2nTX is returned to the internal RLIN2nRX (loop back).

The functions of the LIN self-test mode operate in the following conditions:

- LIN self-test mode (transmission): header transmission and response transmission
- LIN self-test mode (reception): header reception and response reception

In LIN self-test mode, the operation is at the fastest baud rate, regardless of the setting of the baud rate generator. Regardless of the setting of the baud rate related registers, the baud rate operates at the LIN communication clock source/16 [bps].

In LIN self-test mode, the following functions are not supported:

- LIN Wake-up function
- Frame separate mode

Do not use above functions.

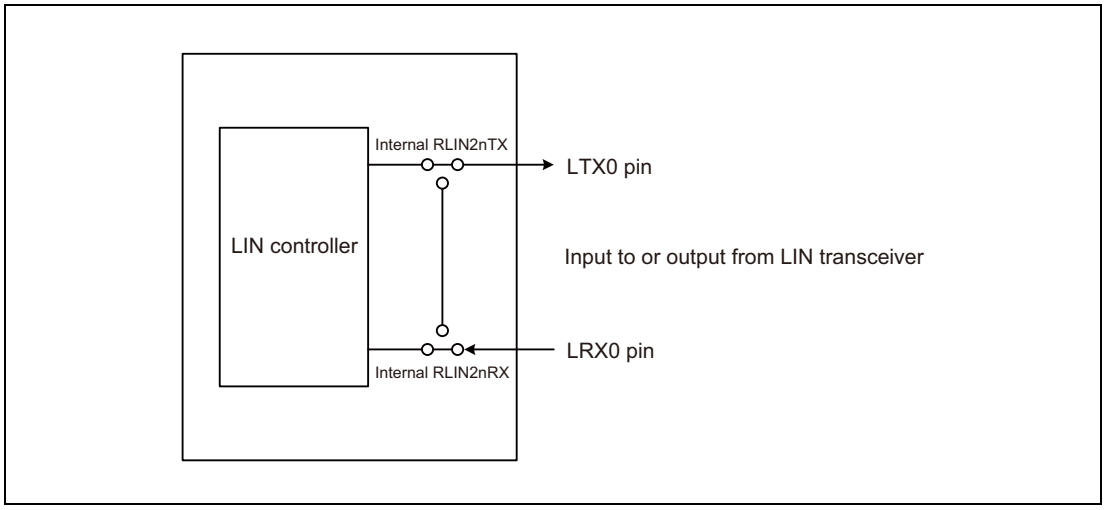


Figure 15.14 Connection in LIN Reset Mode, LIN Wake-Up Mode, and LIN Operation Mode

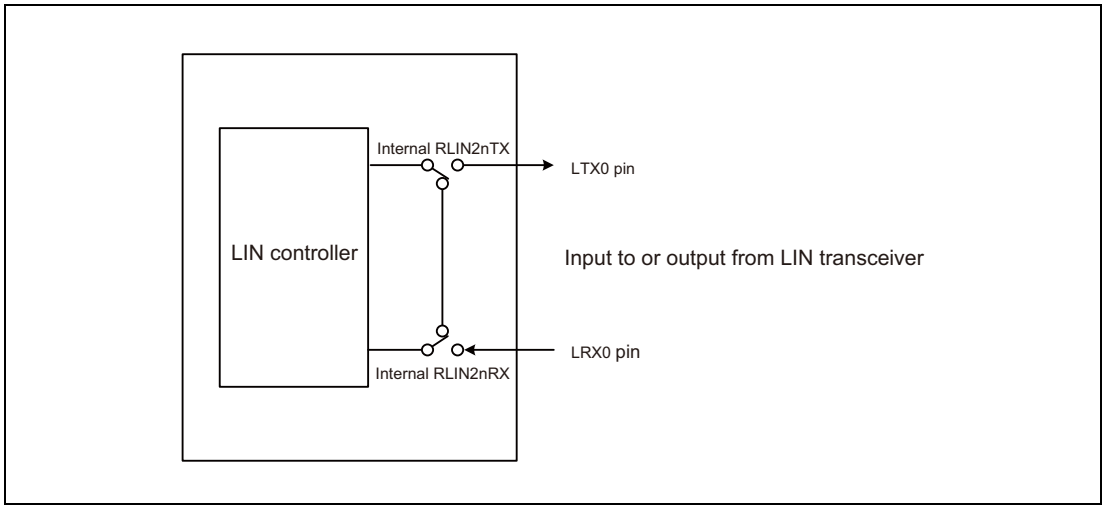


Figure 15.15 Connection in LIN Self-Test Mode

15.15.1 Change to LIN Self-Test Mode

Writing to the RLN21nGLSTC register enables LIN self-test mode.

The RLN21nGLSTC register becoming 01_H indicates that the mode is transit to LIN self-test mode.

When changing to LIN self-test mode, be sure to execute a specific sequence. In that sequence, information must be written three times consecutively to the LIN self-test control register, as follows:

- Change to LIN reset mode regarding all channels in the unit.
  Set the OM0 bit in the RLN21nmLiCUC register to 0 (LIN reset mode).
  Read the OMM0 bit in the RLN21nmLiMST register; verify that it is 0 (LIN reset mode).
- 1st write: RLN21nGLSTC register = $1010\ 0111_B$ ($A7_H$)
- 2nd write: RLN21nGLSTC register = $0101\ 1000_B$ (58_H)
- 3rd write: RLN21nGLSTC register = $0000\ 0001_B$ (01_H)
- Verify the transition to LIN self-test mode
  Read the RLN21nGLSTC register; verify that it is 01_H (LIN self-test mode).

If the key of the first write ($A7_H$) is written twice by mistake, the transition to LIN self-test mode is suspended. The above sequence should be retried from the step of first write. In addition, if a write to another LIN-related register in the same unit is performed during transition to LIN self-test mode (three consecutive write operations to the RLN21nGLSTC register), the transition is also suspended.

15.15.2 Transmission in LIN Self-Test Mode

To execute a self-test on transmission, perform the procedure below:

- Set the baud rate related registers
  RLN21nGLBRP0 register = xxxx $xxxx_B$ *1
  RLN21nGLBRP1 register = xxxx $xxxx_B$ *1
  RLN21nmLiMD register = 0000 $xx00_B$ *1
- Set interrupt enable register and error enable related registers
  RLN21nmLiIE register = 0000 $0xxx_B$ *2
  RLN21nmLiEDE register = 0000 $xxxx_B$
- Set the break field and space related registers.
  RLN21nmLiBFC register = 00xx $xxxx_B$
  RLN21nmLiSC register = 00xx $0xxx_B$
- Release from the LIN reset mode
  Write 11_B to the OM1 and OM0 bits in the RLN21nmLiCUC register, and check that the OMM1 and OMM0 bits in the RLN21nmLiMST register are 11_B.
- Set the transmit frame related registers.
  RLN21nmLiDFC register = 00x1 $xxxx_B$
  RLN21nmLiIDB register = xxxx $xxxx_B$
  RLN21nmLiDBR1 to RLN21nmLiDBR8 registers = xxxx $xxxx_B$
- Header transmission → response transmission started
  Set the FTS bit in the RLN21nmLiTRC register to 1 (frame transmission or wake-up transmission/reception started).
  The LIN self-test mode (transmission) is executed. In this mode, interrupt are generated, and status and error status are also updated. The checksum is automatically calculated by the LIN master interface.
  To suspend the LIN self-test mode (transmission) while it is running, set the OM0 bit in the RLN21nmLiCUC register to 0 (LIN reset mode), which causes a transition to the LIN reset mode.
- When the transmission is completed, the inverse of the looped-back frame data is stored in the RLN21nmLiIDB, RLN21nmLiDBRb, and RLN21nmLiCBR registers (the data is inverted before being stored because the transmitted value should be compared with the looped-back value). Then, the FTS bit in the RLN21nmLiTRC register is cleared.
- If the transmission fails to complete due to an error, the applicable error flag is set and the FTS bit in the RLN21nmLiTRC register is cleared.

NOTE

x: Don't care

Note 1. The following register settings are not reflected to the operation of the LIN self-test mode. The LCKS bit in the RLN21nGLBRP0 register, RLN21nGLBRP1 register, and RLN21nmLiMD register. Therefore, those settings are not necessary.

Note 2. If necessary, set the related registers described in **Section 6, Interrupt**.

15.15.3 Reception in LIN Self-Test Mode

To execute a self-test on LIN master transmission, perform the procedure below:

- Set the baud rate, noise filter, and interrupt output related registers.
  RLN21nGLBRP0 register = xxxx $xxxx_B$ *1
  RLN21nGLBRP1 register = xxxx $xxxx_B$ *1
  RLN21nmLiMD register = 0000 $xx00_B$ *1
- Set the interrupt enable and error enable related registers.
  RLN21nmLiIE register = 0000 $0xxx_B$ *2
  RLN21nmLiEDE register = 0000 $xxxx_B$
- Set the break field and space related registers.
  RLN21nmLiBFC register = 00xx $xxxx_B$
  RLN21nmLiSC register = 00xx $0xxx_B$ *1
- Release from the LIN reset mode
  Write 11_B to the OM1 and OM0 bits in the RLN21nmLiCUC register, and check that the OMM1 and OMM0 bits in the RLN21nmLiMST register are 11_B.
- Set the receive frame related registers.
  RLN21nmLiDFC register = 00x0 $xxxx_B$
  RLN21nmLiIDB register = xxxx $xxxx_B$
  RLN21nmLiDBR1 to RLN21nmLiDBR8 registers = xxxx $xxxx_B$
  RLN21nmLiCBR register = xxxx $xxxx_B$
  Since the checksum value to be transmitted is not automatically calculated, set the calculation value to the RLN21nmLiCBR register. If an incorrect checksum is set at this time, the checksum error can be tested.
- Header transmission → response reception started
  Set the FTS bit in the RLN21nmLiTRC register to 1 (frame transmission or wake-up transmission/reception started).
  The LIN self-test mode (reception) is executed. In this mode, interrupt are generated, and status and error status are also updated.
  To suspend the LIN self-test mode (reception) while it is running, set the OM0 bit in the RLN21nmLiCUC register to 0 (LIN reset mode), which causes a transition to the LIN reset mode.
- When the reception is completed, the inverse of the looped-back frame data is stored in the RLN21nmLiIDB, RLN21nmLiDBRb, and RLN21nmLiCBR registers (the data is inverted before being stored because the set value should be compared with the looped-back value). Then, the FTS bit in the RLN21nmLiTRC register is cleared.
- If the reception fails to complete due to an error, the applicable error flag is set and the FTS bit in the RLN21nmLiTRC register is cleared.

NOTE

x: Don't care

Note 1. The following register settings are not reflected to the operation of the LIN self-test mode. The LCKS bit in the RLN21nGLBRP0 register, the RLN21nGLBRP1 register, and the RLN21nmLiMD register, and the IBS bit and IBHS bit (response space only) in the RLN21nmLiSC register.
Therefore, those settings are not necessary.

Note 2. If necessary, set the related registers described in **Section 6, Interrupt**.

15.15.4 Terminating LIN Self-Test Mode

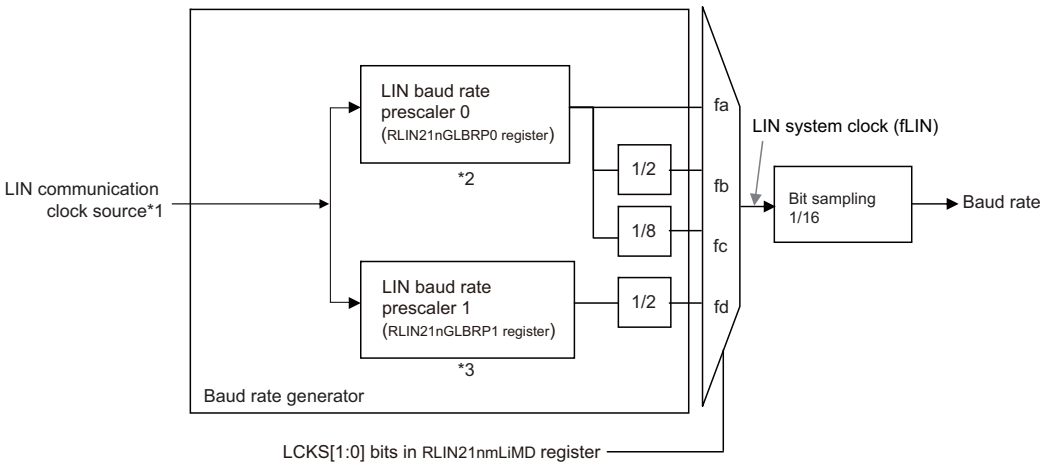
To terminate LIN self-test mode, perform the procedure below:

- All the channels in the unit make a transition to LIN reset mode
  Write 0 to the OM0 bit in the RLN21nmLiCUC register to make a transition to LIN reset mode. However, if the OMM1 and OMM0 bits in the RLN21nmLiMST register are not "11_B" in any channels of the unit after the transition to LIN self-test mode, write "11_B" to the OM1 and OM0 bits in the RLN21nmLiCUC register in any one channel. Check that the OMM1 and OMM0 bits in the RLN21nmLiMST register are set to "11_B", and then make a transition to LIN reset mode.
- Verify release from LIN self-test mode.
  Read the RLN21nGLSTC register; confirm that it is 00_H (not in LIN self-test mode)
- Verify the transition to LIN reset mode.
  Read the OMM0 bit in the RLN21nmLiMST register; verify that it is 0 (LIN reset mode).

15.16 Baud Rate Generator

The LIN system clock (fLIN) is the clock that is made by dividing the LIN communication clock source by the baud rate generator, and the bit rate is made by dividing that clock by 16. The inverse of this bit rate is the bit time (Tbit).

Figure 15.16 shows a block diagram of baud rate generation.



Note 1. For the LIN communication clock source, refer to **Section 11, Clock Controller**
Note 2. According to the RLN21nGLBRP0 register, the clock frequency is divided by N + 1 (N = 0 to 255).
Note 3. According to the RLIN21nGLBPR1 register, the clock frequency is divided by M + 1 (M = 0 to 255).

Figure 15.16 Block Diagram of Baud Rate Generation in LIN Master Mode

Set the LIN communication clock source to the range from 40 MHz.

By setting the RLN21nGLBRP0 register so that fa is 307200 Hz (= 19200 × 16), the resulting bit rates are fa = 19200 × 16, fb = 9600 × 16 and fc = 2400 × 16. These bit rates are frequency-divided by 16 in the bit timing generator, enabling bit rates of 19200 bps, 9600 bps and 2400 bps to be generated. Also, by setting the RLN21nGLBRP1 register so that fd is 166672 Hz (= 10417 × 16), the resulting bit rate is fd = 10417 × 16. This bit rate is frequency-divided by 16 in the bit timing generator, enabling 10417 bps to be generated.

The formula for baud rate is as shown below.

Baud rate:

= {Frequency of LIN communication clock source} / ((RLN21nGLBRP0) + 1) / 16 [bps] (When fa is selected)

= {Frequency of LIN communication clock source} / ((RLN21nGLBRP0) + 1) / 2 / 16 [bps] (When fb is selected)

= {Frequency of LIN communication clock source} / ((RLN21nGLBRP0) + 1) / 8 / 16 [bps] (When fc is selected)

= { Frequency of LIN communication clock source } / ((RLN21nGLBRP1) + 1) / 2 / 16 [bps] (When fd is selected)

Table 15.37 shows examples of baud rate (19200, 10417, 9600, and 2400 bps) generation and also the corresponding errors.

Table 15.37 Examples of Baud Rate (19200, 10417, 9600, and 2400 bps) Generation

LIN Communication Clock Source	Baud Rate Generator 0 (N+1) Frequency-Divided	Baud Rate Generator 1 (M+1) Frequency-Divided	System Clock	Baud Rate	Error
40MHz	130	—	fa	19230.77	+0.16%
	—	120	fd	10416.67	-0.003%
	130	—	fb	9615.38	+0.16%
	130	—	fc	2403.85	+0.16%

Section 16 CAN Interface (RS-CAN)

16.1 Overview of RS-CAN

16.1.1 Number of Units

This microcontroller incorporates the following number of units of the CAN interface (RS-CAN).

Table 16.1 Number of Units

	RS-CAN
Number of units	1
Name	RS-CANn (n = 0)

Table 16.2 Index

Index	
n	In this section, the unit of RS-CAN is identified by “n” (n = 0). For example, a global control register of RS-CAN0 is written as RSCAN0GCTR.
m	In this section, the channel number of RS-CAN is identified by “m” (m = 0 to 3). For example, a channel m status register is written as RSCAN0mSTS (m = 0 to 3).
j	In this section, a register related to reception rule table is identified by “j” (j = 0 to 15). For example, a reception rule ID register is written as RSCAN0GAFLIDj (j = 0 to 15).
k	In this section, the number of transmit/reception FIFO buffer is identified by “k” (k = 0 to 11). For example, a transmit/reception FIFO buffer configuration/control register is written as RSCAN0CFCCk (k = 0 to 14).
x	In this section, the number of reception FIFO is identified by “x” (x = 0 to 7). For example, a reception FIFO buffer status register is written as RSCAN0RFSTSx (x = 0 to 7).
q	In this section, the number of reception buffer is identified by “q” (q = 0 to 63). For example, a reception buffer ID register is written as RSCAN0RMIDq (q = 0 to 63).
p	In this section, the number of transmission buffer is identified by “p” (p = 0 to 63). For example, a transmission buffer control register is written as RSCAN0TMCp (p = 0 to 63).
r	In this section, the number of RAM test for CAN is identified by “r” (r = 0 to 63). For example, a RAM test page access register is written as RSCAN0RPGACCr (r = 0 to 63).
y	In another case, y is used to identify. For example, a reception buffer new data register is written as RSCAN0RMNDy (y = 0 to 1).

The RS-CAN has four channels.

Table 16.3 Number of Channels

Unit Name	Number of Channels	Channel Name
RS-CAN0	4	CANm (m = 0 to 3)

16.1.2 Register Base Address

The base addresses of registers in RS-CAN are described in the list below.

All Access to RS-CAN register addresses are given as address offsets to the base address.

Table 16.4 Register Base Address

Base Address Name	Base Address
<RS-CAN0_base>	FFD0 0000_H

16.1.3 Clock Supply

The clocks listed below are supplied to the RS-CAN.

Table 16.5 RS-CAN Clock Supply

Unit	Clock for the Unit	Clock Supply Name
RS-CAN0	pclk	CLK_HSB (High speed peripheral clock)
	clkc	CLKC_LSB (Unmodulated low speed peripheral clock)
	clk_xincan	Main OSC

16.1.4 Interrupt and DMA / DTS

RS-CAN can generate the interrupt requests described in the list below.

Table 16.6 Interrupt Request

Interrupt Name	Functions	Number of Interrupts
Global interrupt		
—	Receive FIFOm interrupt	412
—	Global error interrupt	413
Channel CAN0 interrupt		
—	CAN0 transmit/receive FIFO successful reception interrupt (when reception mode, gateway mode)	414
—	CAN0 error interrupt	415
—	CAN0 transmission interrupt	416
Channel CAN1 interrupt		
—	CAN1 transmit/receive FIFO successful reception interrupt (when reception mode, gateway mode)	417
—	CAN1 error interrupt	418
—	CAN1 transmission interrupt	419
Channel CAN2 interrupt		
—	CAN2 transmit/receive FIFO successful reception interrupt (when reception mode, gateway mode)	420
—	CAN2 error interrupt	421
—	CAN2 transmission interrupt	422
Channel CAN3 interrupt		
—	CAN3 transmit/receive FIFO successful reception interrupt (when reception mode, gateway mode)	423
—	CAN3 error interrupt	424
—	CAN3 transmission interrupt	425

16.1.5 Reset Sources

RS-CAN is initialized by the reset sources described in the following table.

Table 16.7 Reset Sources

Unit name	Reset Sources
RS-CAN0	All reset sources

16.1.6 External I/O Signals

The external I/O signals of RS-CAN are listed below.

Table 16.8 External I/O Signals

Unit Signal Name	Features	Signal Names Shared with Port Pins
Rxm (m = 0 to 3)	Receive data input pin of CAN communication function	CRX0, CRX1, CRX2, CRX3
Txm (m = 0 to 3)	Transmit data output pin of CAN communication function	CTX0, CTX1, CTX2, CTX3

16.2 Overview

16.2.1 Functional Outline

The RH850/E1M-S incorporates one unit of CAN interface (RS-CAN) mounting four channels (CAN0 to CAN3) of the Controller Area Network (RS-CAN) module conforming to the ISO11898-1 specifications.

Table 16.9 shows the RS-CAN module specifications. **Figure 16.1** shows the RS-CAN module block diagram.

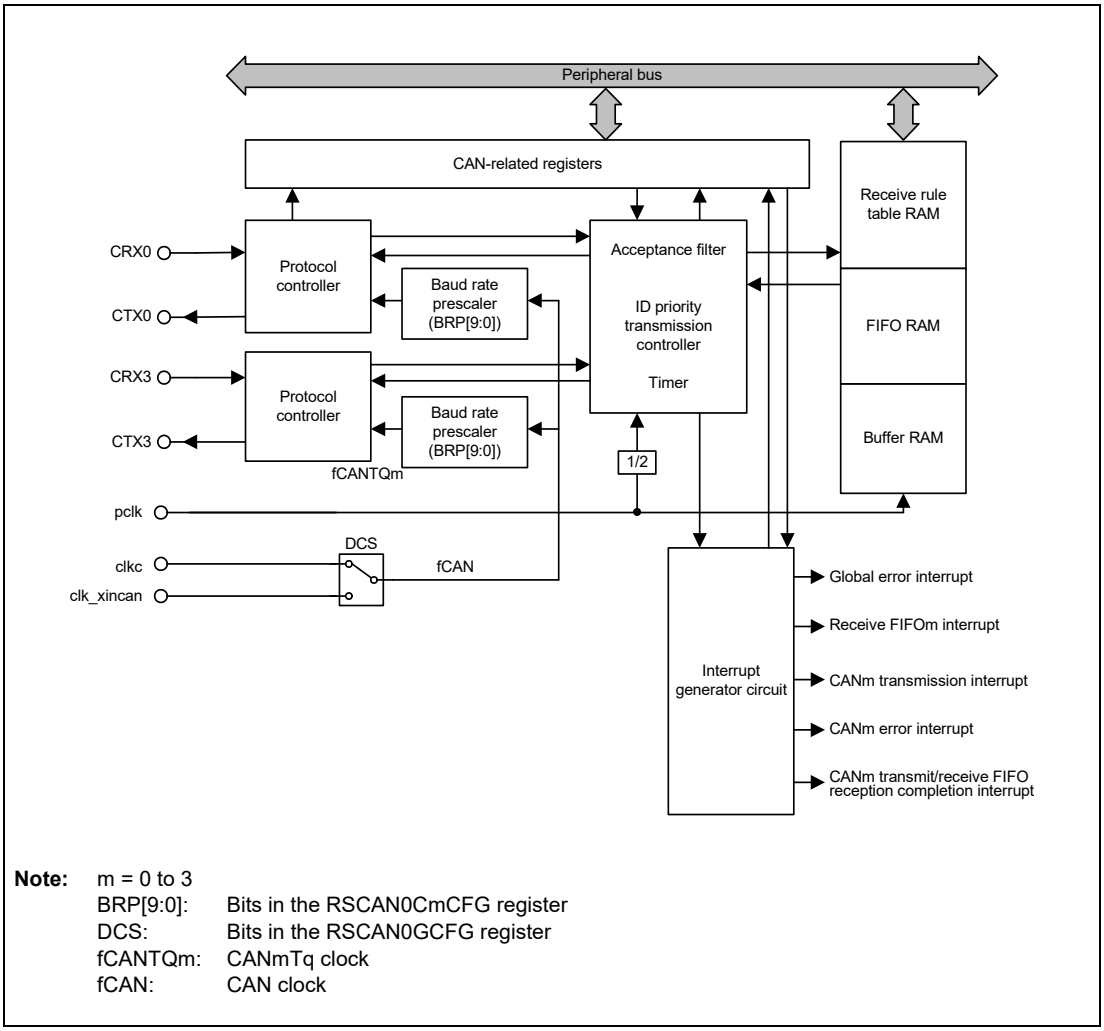
Table 16.9 RS-CAN Module Specifications (1/2)

Item	Specification
Number of channels	4
Protocol	ISO11898-1 compliant
Communication speed	• Up to 1 Mbps $\text{Communication speed (CANn bit time clock)} = \frac{1}{\text{CANm bit time}}$ CANm bit time = CANmTq × Tq count per bit $\text{CANmTq} = \frac{\text{(BRP[9:0] bits in the RCAN0mCFG register + 1)}}{\text{fCAN}}$ m = 0 to 3 Tq: Time quantum fCAN: Frequency of CAN clock (selected by the DCS bit in the RSCAN0GCFG register)
Buffer	320 buffers in total • Individual buffers: 64 buffers (16 buffers ×4 channels) Transmit buffer: 16 buffers per channel Transmit queue: Single queue per channel (shared with the transmit buffer; up to 16 buffers allocatable) • Shared buffers: 256 buffers for all channels Receive buffer: 0 to 64 buffers for all channels Receive FIFO buffer: 8 FIFO buffers for all channels (up to 128 buffers allocatable to each) Transmit/receive FIFO buffer: 3 FIFO buffers per channel (up to 128 buffers allocatable to each)
Receive function	• Receives data frames and remote frames. • Selects ID format (standard ID, extended ID, or both IDs) to be received. • Sets interrupt enable/disable for each FIFO. • Mirror function (CAN mode receives its own transmitted message.) • Timestamp function (to record message reception time as a 16-bit timer value)
Receive filter function	• Selects receive messages according to 256 receive rules. • Sets the number of receive rules (0 to 128) for each channel. • Acceptance filter processing: Sets ID and mask for each receive rule. • DLC filter processing: Sets DLC check value for each acceptance rule.
Receive message transfer function	• Routing function to transfer receive messages to arbitrary destinations (can be transferred to up to 8 buffers) Transfer destination: Receive buffer, receive FIFO buffer, and/or transmit/receive FIFO buffer • Label addition function Stores label information together with the message in a receive buffer and FIFO buffer.
Transmission function	• Transmits data frames and remote frames. • Selects ID format (standard ID, extended ID, or both IDs) to be transmitted. • Sets interrupt enable/disable for each transmit buffer and transmit/receive FIFO buffer. • Selects ID priority transmission or transmit buffer number priority transmission. • Transmit request can be aborted (possible to confirm with a flag) • One-shot transmission function

Table 16.9 RS-CAN Module Specifications (2/2)

Item	Specification
Interval transmission function	Transmit messages at intervals (transmit mode or gateway mode of transmit/receive FIFO buffers)
Transmit queue function	Transmits all stored messages according to the ID priority.
Transmit history function	Stores the history information of transmitted messages.
Gateway function	A received message is automatically routed to a different channel.
Bus off recovery mode selection	Selects the method for returning from bus off state. • ISO11898-1 compliant • Automatic entry to channel halt mode at bus-off entry • Automatic entry to channel halt mode at bus-off end • Entry to channel halt mode by program request • Transition to the error-active state by program request (forcible return from bus-off)
Error status monitoring	• Monitors CAN protocol errors (stuff error, form error, ACK error, CRC error, bit error, ACK delimiter error, and bus dominant lock). • Detects error status transitions (error warning, error passive, bus off entry, and bus off recovery) • Reads the error counter. • Monitors DLC errors.
Interrupt source	14 sources • Global Interrupts (2 sources) Receive FIFO interrupt Global error interrupt • Channel interrupts (3 sources/channel) CANm transmission interrupt (m = 0 to 3) – CANm transmission complete interrupt – CANm transmission abort interrupt – CANm transmit/receive FIFO transmission complete interrupt (in transmit mode, gateway mode) – CANm transmit history interrupt – CANm transmit queue interrupt CANm transmit/receive FIFO reception complete interrupt (in receive mode, gateway mode) CANm error interrupt
CAN stop mode	Reduces power consumption by stopping clock supply to the RS-CAN module.
CAN clock source	Clkc or clk_xincan selectable
Test function	Test function for user evaluation • Listen-only mode • Self-test mode 0 (external loopback) • Self-test mode 1 (internal loopback) • RAM test (read/write test) • Inter-channel communication test

16.2.2 Block Diagram



Note: m = 0 to 3
BRP[9:0]: Bits in the RSCAN0CmCFG register
DCS: Bits in the RSCAN0GCFG register
fCANTQm: CANmTq clock
fCAN: CAN clock

Figure 16.1 RS-CAN Module Block Diagram

16.3 Register Descriptions

16.3.1 Register List

RS-CAN is controlled and operated by the following registers.

Table 16.10 List of RS-CAN Module Registers (1/21)

Register Name	Symbol	Value after Reset	Address	Access Size
Channel 0 configuration register	RSCAN0C0CFG	0000 0000$_H$	<RS-CAN0_base> + 0000$_H$	8, 16, 32
Channel 0 control register	RSCAN0C0CTR	0000 0005$_H$	<RS-CAN0_base> + 0004$_H$	8, 16, 32
Channel 0 status register	RSCAN0C0STS	0000 0005$_H$	<RS-CAN0_base> + 0008$_H$	8, 16, 32
Channel 0 error flag register	RSCAN0C0ERFL	0000 0000$_H$	<RS-CAN0_base> + 000C$_H$	8, 16, 32
Channel 1 configuration register	RSCAN0C1CFG	0000 0000$_H$	<RS-CAN0_base> + 0010$_H$	8, 16, 32
Channel 1 control register	RSCAN0C1CTR	0000 0005$_H$	<RS-CAN0_base> + 0014$_H$	8, 16, 32
Channel 1 status register	RSCAN0C1STS	0000 0005$_H$	<RS-CAN0_base> + 0018$_H$	8, 16, 32
Channel 1 error flag register	RSCAN0C1ERFL	0000 0000$_H$	<RS-CAN0_base> + 001C$_H$	8, 16, 32
Channel 2 configuration register	RSCAN0C2CFG	0000 0000$_H$	<RS-CAN0_base> + 0020$_H$	8, 16, 32
Channel 2 control register	RSCAN0C2CTR	0000 0005$_H$	<RS-CAN0_base> + 0024$_H$	8, 16, 32
Channel 2 status register	RSCAN0C2STS	0000 0005$_H$	<RS-CAN0_base> + 0028$_H$	8, 16, 32
Channel 2 error flag register	RSCAN0C2ERFL	0000 0000$_H$	<RS-CAN0_base> + 002C$_H$	8, 16, 32
Channel 3 configuration register	RSCAN0C3CFG	0000 0000$_H$	<RS-CAN0_base> + 0030$_H$	8, 16, 32
Channel 3 control register	RSCAN0C3CTR	0000 0005$_H$	<RS-CAN0_base> + 0034$_H$	8, 16, 32
Channel 3 status register	RSCAN0C3STS	0000 0005$_H$	<RS-CAN0_base> + 0038$_H$	8, 16, 32
Channel 3 error flag register	RSCAN0C3ERFL	0000 0000$_H$	<RS-CAN0_base> + 003C$_H$	8, 16, 32
Global configuration register	RSCAN0GCFG	0000 0000$_H$	<RS-CAN0_base> + 0084$_H$	8, 16, 32
Global control register	RSCAN0GCTR	0000 0005$_H$	<RS-CAN0_base> + 0088$_H$	8, 16, 32
Global status register	RSCAN0GSTS	0000 000D$_H$	<RS-CAN0_base> + 008C$_H$	8, 16, 32
Global error flag register	RSCAN0GERFL	0000 0000$_H$	<RS-CAN0_base> + 0090$_H$	8, 16, 32
Global timestamp counter register	RSCAN0GTSC	0000 0000$_H$	<RS-CAN0_base> + 0094$_H$	16, 32
Global acceptance filter list entry control register	RSCAN0GAFLECTR	0000 0000$_H$	<RS-CAN0_base> + 0098$_H$	8, 16, 32
Global acceptance filter list configuration register 0	RSCAN0GAFLCFG0	0000 0000$_H$	<RS-CAN0_base> + 009C$_H$	8, 16, 32
Receive buffer number register	RSCAN0RMNB	0000 0000$_H$	<RS-CAN0_base> + 00A4$_H$	8, 16, 32
Receive buffer new data register 0	RSCAN0RMND0	0000 0000$_H$	<RS-CAN0_base> + 00A8$_H$	8, 16, 32
Receive buffer new data register 1	RSCAN0RMND1	0000 0000$_H$	<RS-CAN0_base> + 00AC$_H$	8, 16, 32
Receive FIFO buffer configuration/control register 0	RSCAN0RFCC0	0000 0000$_H$	<RS-CAN0_base> + 00B8$_H$	8, 16, 32
Receive FIFO buffer configuration/control register 1	RSCAN0RFCC1	0000 0000$_H$	<RS-CAN0_base> + 00BC$_H$	8, 16, 32
Receive FIFO buffer configuration/control register 2	RSCAN0RFCC2	0000 0000$_H$	<RS-CAN0_base> + 00C0$_H$	8, 16, 32
Receive FIFO buffer configuration/control register 3	RSCAN0RFCC3	0000 0000$_H$	<RS-CAN0_base> + 00C4$_H$	8, 16, 32
Receive FIFO buffer configuration/control register 4	RSCAN0RFCC4	0000 0000$_H$	<RS-CAN0_base> + 00C8$_H$	8, 16, 32
Receive FIFO buffer configuration/control register 5	RSCAN0RFCC5	0000 0000$_H$	<RS-CAN0_base> + 00CC$_H$	8, 16, 32
Receive FIFO buffer configuration/control register 6	RSCAN0RFCC6	0000 0000$_H$	<RS-CAN0_base> + 00D0$_H$	8, 16, 32
Receive FIFO buffer configuration/control register 7	RSCAN0RFCC7	0000 0000$_H$	<RS-CAN0_base> + 00D4$_H$	8, 16, 32
Receive FIFO buffer status register 0	RSCAN0RFSTS0	0000 0001$_H$	<RS-CAN0_base> + 00D8$_H$	8, 16, 32
Receive FIFO buffer status register 1	RSCAN0RFSTS1	0000 0001$_H$	<RS-CAN0_base> + 00DC$_H$	8, 16, 32
Receive FIFO buffer status register 2	RSCAN0RFSTS2	0000 0001$_H$	<RS-CAN0_base> + 00E0$_H$	8, 16, 32
Receive FIFO buffer status register 3	RSCAN0RFSTS3	0000 0001$_H$	<RS-CAN0_base> + 00E4$_H$	8, 16, 32
Receive FIFO buffer status register 4	RSCAN0RFSTS4	0000 0001$_H$	<RS-CAN0_base> + 00E8$_H$	8, 16, 32
Receive FIFO buffer status register 5	RSCAN0RFSTS5	0000 0001$_H$	<RS-CAN0_base> + 00EC$_H$	8, 16, 32
Receive FIFO buffer status register 6	RSCAN0RFSTS6	0000 0001$_H$	<RS-CAN0_base> + 00F0$_H$	8, 16, 32
Receive FIFO buffer status register 7	RSCAN0RFSTS7	0000 0001$_H$	<RS-CAN0_base> + 00F4$_H$	8, 16, 32
Receive FIFO buffer pointer control register 0	RSCAN0RFPCTR0	0000 0000$_H$	<RS-CAN0_base> + 00F8$_H$	8, 16, 32

Table 16.10 List of RS-CAN Module Registers (2/21)

Register Name	Symbol	Value after Reset	Address	Access Size
Receive FIFO buffer pointer control register 1	RSCAN0RFPCTR1	0000 0000$_H$	<RS-CAN0_base> + 00FC$_H$	8, 16, 32
Receive FIFO buffer pointer control register 2	RSCAN0RFPCTR2	0000 0000$_H$	<RS-CAN0_base> + 0100$_H$	8, 16, 32
Receive FIFO buffer pointer control register 3	RSCAN0RFPCTR3	0000 0000$_H$	<RS-CAN0_base> + 0104$_H$	8, 16, 32
Receive FIFO buffer pointer control register 4	RSCAN0RFPCTR4	0000 0000$_H$	<RS-CAN0_base> + 0108$_H$	8, 16, 32
Receive FIFO buffer pointer control register 5	RSCAN0RFPCTR5	0000 0000$_H$	<RS-CAN0_base> + 010C$_H$	8, 16, 32
Receive FIFO buffer pointer control register 6	RSCAN0RFPCTR6	0000 0000$_H$	<RS-CAN0_base> + 0110$_H$	8, 16, 32
Receive FIFO buffer pointer control register 7	RSCAN0RFPCTR7	0000 0000$_H$	<RS-CAN0_base> + 0114$_H$	8, 16, 32
Transmit/receive FIFO buffer configuration/control register 0	RSCAN0CFCC0	0000 0000$_H$	<RS-CAN0_base> + 0118$_H$	8, 16, 32
Transmit/receive FIFO buffer configuration/control register 1	RSCAN0CFCC1	0000 0000$_H$	<RS-CAN0_base> + 011C$_H$	8, 16, 32
Transmit/receive FIFO buffer configuration/control register 2	RSCAN0CFCC2	0000 0000$_H$	<RS-CAN0_base> + 0120$_H$	8, 16, 32
Transmit/receive FIFO buffer configuration/control register 3	RSCAN0CFCC3	0000 0000$_H$	<RS-CAN0_base> + 0124$_H$	8, 16, 32
Transmit/receive FIFO buffer configuration/control register 4	RSCAN0CFCC4	0000 0000$_H$	<RS-CAN0_base> + 0128$_H$	8, 16, 32
Transmit/receive FIFO buffer configuration/control register 5	RSCAN0CFCC5	0000 0000$_H$	<RS-CAN0_base> + 012C$_H$	8, 16, 32
Transmit/receive FIFO buffer configuration/control register 6	RSCAN0CFCC6	0000 0000$_H$	<RS-CAN0_base> + 0130$_H$	8, 16, 32
Transmit/receive FIFO buffer configuration/control register 7	RSCAN0CFCC7	0000 0000$_H$	<RS-CAN0_base> + 0134$_H$	8, 16, 32
Transmit/receive FIFO buffer configuration/control register 8	RSCAN0CFCC8	0000 0000$_H$	<RS-CAN0_base> + 0138$_H$	8, 16, 32
Transmit/receive FIFO buffer configuration/control register 9	RSCAN0CFCC9	0000 0000$_H$	<RS-CAN0_base> + 013C$_H$	8, 16, 32
Transmit/receive FIFO buffer configuration/control register 10	RSCAN0CFCC10	0000 0000$_H$	<RS-CAN0_base> + 0140$_H$	8, 16, 32
Transmit/receive FIFO buffer configuration/control register 11	RSCAN0CFCC11	0000 0000$_H$	<RS-CAN0_base> + 0144$_H$	8, 16, 32
Transmit/receive FIFO buffer status register 0	RSCAN0CFSTS0	0000 0001$_H$	<RS-CAN0_base> + 0178$_H$	8, 16, 32
Transmit/receive FIFO buffer status register 1	RSCAN0CFSTS1	0000 0001$_H$	<RS-CAN0_base> + 017C$_H$	8, 16, 32
Transmit/receive FIFO buffer status register 2	RSCAN0CFSTS2	0000 0001$_H$	<RS-CAN0_base> + 0180$_H$	8, 16, 32
Transmit/receive FIFO buffer status register 3	RSCAN0CFSTS3	0000 0001$_H$	<RS-CAN0_base> + 0184$_H$	8, 16, 32
Transmit/receive FIFO buffer status register 4	RSCAN0CFSTS4	0000 0001$_H$	<RS-CAN0_base> + 0188$_H$	8, 16, 32
Transmit/receive FIFO buffer status register 5	RSCAN0CFSTS5	0000 0001$_H$	<RS-CAN0_base> + 018C$_H$	8, 16, 32
Transmit/receive FIFO buffer status register 6	RSCAN0CFSTS6	0000 0001$_H$	<RS-CAN0_base> + 0190$_H$	8, 16, 32
Transmit/receive FIFO buffer status register 7	RSCAN0CFSTS7	0000 0001$_H$	<RS-CAN0_base> + 0194$_H$	8, 16, 32
Transmit/receive FIFO buffer status register 8	RSCAN0CFSTS8	0000 0001$_H$	<RS-CAN0_base> + 0198$_H$	8, 16, 32
Transmit/receive FIFO buffer status register 9	RSCAN0CFSTS9	0000 0001$_H$	<RS-CAN0_base> + 019C$_H$	8, 16, 32
Transmit/receive FIFO buffer status register 10	RSCAN0CFSTS10	0000 0001$_H$	<RS-CAN0_base> + 01A0$_H$	8, 16, 32
Transmit/receive FIFO buffer status register 11	RSCAN0CFSTS11	0000 0001$_H$	<RS-CAN0_base> + 01A4$_H$	8, 16, 32
Transmit/receive FIFO buffer pointer control register 0	RSCAN0CFPCTR0	0000 0000$_H$	<RS-CAN0_base> + 01D8$_H$	8, 16, 32
Transmit/receive FIFO buffer pointer control register 1	RSCAN0CFPCTR1	0000 0000$_H$	<RS-CAN0_base> + 01DC$_H$	8, 16, 32
Transmit/receive FIFO buffer pointer control register 2	RSCAN0CFPCTR2	0000 0000$_H$	<RS-CAN0_base> + 01E0$_H$	8, 16, 32
Transmit/receive FIFO buffer pointer control register 3	RSCAN0CFPCTR3	0000 0000$_H$	<RS-CAN0_base> + 01E4$_H$	8, 16, 32
Transmit/receive FIFO buffer pointer control register 4	RSCAN0CFPCTR4	0000 0000$_H$	<RS-CAN0_base> + 01E8$_H$	8, 16, 32
Transmit/receive FIFO buffer pointer control register 5	RSCAN0CFPCTR5	0000 0000$_H$	<RS-CAN0_base> + 01EC$_H$	8, 16, 32
Transmit/receive FIFO buffer pointer control register 6	RSCAN0CFPCTR6	0000 0000$_H$	<RS-CAN0_base> + 01F0$_H$	8, 16, 32
Transmit/receive FIFO buffer pointer control register 7	RSCAN0CFPCTR7	0000 0000$_H$	<RS-CAN0_base> + 01F4$_H$	8, 16, 32
Transmit/receive FIFO buffer pointer control register 8	RSCAN0CFPCTR8	0000 0000$_H$	<RS-CAN0_base> + 01F8$_H$	8, 16, 32
Transmit/receive FIFO buffer pointer control register 9	RSCAN0CFPCTR9	0000 0000$_H$	<RS-CAN0_base> + 01FC$_H$	8, 16, 32
Transmit/receive FIFO buffer pointer control register 10	RSCAN0CFPCTR10	0000 0000$_H$	<RS-CAN0_base> + 0200$_H$	8, 16, 32
Transmit/receive FIFO buffer pointer control register 11	RSCAN0CFPCTR11	0000 0000$_H$	<RS-CAN0_base> + 0204$_H$	8, 16, 32
FIFO empty status register	RSCAN0FESTS	007F FFFF$_H$	<RS-CAN0_base> + 0238$_H$	8, 16, 32
FIFO full status register	RSCAN0FFSTS	0000 0000$_H$	<RS-CAN0_base> + 023C$_H$	8, 16, 32
FIFO message lost status register	RSCAN0FMSTS	0000 0000$_H$	<RS-CAN0_base> + 0240$_H$	8, 16, 32
Receive FIFO buffer interrupt flag status register	RSCAN0RFISTS	0000 0000$_H$	<RS-CAN0_base> + 0244$_H$	8, 16, 32
Transmit/receive FIFO buffer reception interrupt flag status register	RSCAN0CFRISTS	0000 0000$_H$	<RS-CAN0_base> + 0248$_H$	8, 16, 32

Table 16.10 List of RS-CAN Module Registers (3/21)

Register Name	Symbol	Value after Reset	Address	Access Size
Transmit/receive FIFO buffer transmission interrupt flag status register	RSCAN0CFTISTS	0000 0000_H	<RS-CAN0_base> + $024C_H$	8, 16, 32
Transmit buffer control register 0	RSCAN0TMC0	00_H	<RS-CAN0_base> + 0250_H	8
Transmit buffer control register 1	RSCAN0TMC1	00_H	<RS-CAN0_base> + 0251_H	8
Transmit buffer control register 2	RSCAN0TMC2	00_H	<RS-CAN0_base> + 0252_H	8
Transmit buffer control register 3	RSCAN0TMC3	00_H	<RS-CAN0_base> + 0253_H	8
Transmit buffer control register 4	RSCAN0TMC4	00_H	<RS-CAN0_base> + 0254_H	8
Transmit buffer control register 5	RSCAN0TMC5	00_H	<RS-CAN0_base> + 0255_H	8
Transmit buffer control register 6	RSCAN0TMC6	00_H	<RS-CAN0_base> + 0256_H	8
Transmit buffer control register 7	RSCAN0TMC7	00_H	<RS-CAN0_base> + 0257_H	8
Transmit buffer control register 8	RSCAN0TMC8	00_H	<RS-CAN0_base> + 0258_H	8
Transmit buffer control register 9	RSCAN0TMC9	00_H	<RS-CAN0_base> + 0259_H	8
Transmit buffer control register 10	RSCAN0TMC10	00_H	<RS-CAN0_base> + $025A_H$	8
Transmit buffer control register 11	RSCAN0TMC11	00_H	<RS-CAN0_base> + $025B_H$	8
Transmit buffer control register 12	RSCAN0TMC12	00_H	<RS-CAN0_base> + $025C_H$	8
Transmit buffer control register 13	RSCAN0TMC13	00_H	<RS-CAN0_base> + $025D_H$	8
Transmit buffer control register 14	RSCAN0TMC14	00_H	<RS-CAN0_base> + $025E_H$	8
Transmit buffer control register 15	RSCAN0TMC15	00_H	<RS-CAN0_base> + $025F_H$	8
Transmit buffer control register 16	RSCAN0TMC16	00_H	<RS-CAN0_base> + 0260_H	8
Transmit buffer control register 17	RSCAN0TMC17	00_H	<RS-CAN0_base> + 0261_H	8
Transmit buffer control register 18	RSCAN0TMC18	00_H	<RS-CAN0_base> + 0262_H	8
Transmit buffer control register 19	RSCAN0TMC19	00_H	<RS-CAN0_base> + 0263_H	8
Transmit buffer control register 20	RSCAN0TMC20	00_H	<RS-CAN0_base> + 0264_H	8
Transmit buffer control register 21	RSCAN0TMC21	00_H	<RS-CAN0_base> + 0265_H	8
Transmit buffer control register 22	RSCAN0TMC22	00_H	<RS-CAN0_base> + 0266_H	8
Transmit buffer control register 23	RSCAN0TMC23	00_H	<RS-CAN0_base> + 0267_H	8
Transmit buffer control register 24	RSCAN0TMC24	00_H	<RS-CAN0_base> + 0268_H	8
Transmit buffer control register 25	RSCAN0TMC25	00_H	<RS-CAN0_base> + 0269_H	8
Transmit buffer control register 26	RSCAN0TMC26	00_H	<RS-CAN0_base> + $026A_H$	8
Transmit buffer control register 27	RSCAN0TMC27	00_H	<RS-CAN0_base> + $026B_H$	8
Transmit buffer control register 28	RSCAN0TMC28	00_H	<RS-CAN0_base> + $026C_H$	8
Transmit buffer control register 29	RSCAN0TMC29	00_H	<RS-CAN0_base> + $026D_H$	8
Transmit buffer control register 30	RSCAN0TMC30	00_H	<RS-CAN0_base> + $026E_H$	8
Transmit buffer control register 31	RSCAN0TMC31	00_H	<RS-CAN0_base> + $026F_H$	8
Transmit buffer control register 32	RSCAN0TMC32	00_H	<RS-CAN0_base> + 0270_H	8
Transmit buffer control register 33	RSCAN0TMC33	00_H	<RS-CAN0_base> + 0271_H	8
Transmit buffer control register 34	RSCAN0TMC34	00_H	<RS-CAN0_base> + 0272_H	8
Transmit buffer control register 35	RSCAN0TMC35	00_H	<RS-CAN0_base> + 0273_H	8
Transmit buffer control register 36	RSCAN0TMC36	00_H	<RS-CAN0_base> + 0274_H	8
Transmit buffer control register 37	RSCAN0TMC37	00_H	<RS-CAN0_base> + 0275_H	8
Transmit buffer control register 38	RSCAN0TMC38	00_H	<RS-CAN0_base> + 0276_H	8
Transmit buffer control register 39	RSCAN0TMC39	00_H	<RS-CAN0_base> + 0277_H	8
Transmit buffer control register 40	RSCAN0TMC40	00_H	<RS-CAN0_base> + 0278_H	8
Transmit buffer control register 41	RSCAN0TMC41	00_H	<RS-CAN0_base> + 0279_H	8
Transmit buffer control register 42	RSCAN0TMC42	00_H	<RS-CAN0_base> + $027A_H$	8
Transmit buffer control register 43	RSCAN0TMC43	00_H	<RS-CAN0_base> + $027B_H$	8
Transmit buffer control register 44	RSCAN0TMC44	00_H	<RS-CAN0_base> + $027C_H$	8
Transmit buffer control register 45	RSCAN0TMC45	00_H	<RS-CAN0_base> + $027D_H$	8
Transmit buffer control register 46	RSCAN0TMC46	00_H	<RS-CAN0_base> + $027E_H$	8
Transmit buffer control register 47	RSCAN0TMC47	00_H	<RS-CAN0_base> + $027F_H$	8

Table 16.10 List of RS-CAN Module Registers (4/21)

Register Name	Symbol	Value after Reset	Address	Access Size
Transmit buffer control register 48	RSCAN0TMC48	00_H	<RS-CAN0_base> + 0280_H	8
Transmit buffer control register 49	RSCAN0TMC49	00_H	<RS-CAN0_base> + 0281_H	8
Transmit buffer control register 50	RSCAN0TMC50	00_H	<RS-CAN0_base> + 0282_H	8
Transmit buffer control register 51	RSCAN0TMC51	00_H	<RS-CAN0_base> + 0283_H	8
Transmit buffer control register 52	RSCAN0TMC52	00_H	<RS-CAN0_base> + 0284_H	8
Transmit buffer control register 53	RSCAN0TMC53	00_H	<RS-CAN0_base> + 0285_H	8
Transmit buffer control register 54	RSCAN0TMC54	00_H	<RS-CAN0_base> + 0286_H	8
Transmit buffer control register 55	RSCAN0TMC55	00_H	<RS-CAN0_base> + 0287_H	8
Transmit buffer control register 56	RSCAN0TMC56	00_H	<RS-CAN0_base> + 0288_H	8
Transmit buffer control register 57	RSCAN0TMC57	00_H	<RS-CAN0_base> + 0289_H	8
Transmit buffer control register 58	RSCAN0TMC58	00_H	<RS-CAN0_base> + $028A_H$	8
Transmit buffer control register 59	RSCAN0TMC59	00_H	<RS-CAN0_base> + $028B_H$	8
Transmit buffer control register 60	RSCAN0TMC60	00_H	<RS-CAN0_base> + $028C_H$	8
Transmit buffer control register 61	RSCAN0TMC61	00_H	<RS-CAN0_base> + $028D_H$	8
Transmit buffer control register 62	RSCAN0TMC62	00_H	<RS-CAN0_base> + $028E_H$	8
Transmit buffer control register 63	RSCAN0TMC63	00_H	<RS-CAN0_base> + $028F_H$	8
Transmit buffer status register 0	RSCAN0TMSTS0	00_H	<RS-CAN0_base> + $02D0_H$	8
Transmit buffer status register 1	RSCAN0TMSTS1	00_H	<RS-CAN0_base> + $02D1_H$	8
Transmit buffer status register 2	RSCAN0TMSTS2	00_H	<RS-CAN0_base> + $02D2_H$	8
Transmit buffer status register 3	RSCAN0TMSTS3	00_H	<RS-CAN0_base> + $02D3_H$	8
Transmit buffer status register 4	RSCAN0TMSTS4	00_H	<RS-CAN0_base> + $02D4_H$	8
Transmit buffer status register 5	RSCAN0TMSTS5	00_H	<RS-CAN0_base> + $02D5_H$	8
Transmit buffer status register 6	RSCAN0TMSTS6	00_H	<RS-CAN0_base> + $02D6_H$	8
Transmit buffer status register 7	RSCAN0TMSTS7	00_H	<RS-CAN0_base> + $02D7_H$	8
Transmit buffer status register 8	RSCAN0TMSTS8	00_H	<RS-CAN0_base> + $02D8_H$	8
Transmit buffer status register 9	RSCAN0TMSTS9	00_H	<RS-CAN0_base> + $02D9_H$	8
Transmit buffer status register 10	RSCAN0TMSTS10	00_H	<RS-CAN0_base> + $02DA_H$	8
Transmit buffer status register 11	RSCAN0TMSTS11	00_H	<RS-CAN0_base> + $02DB_H$	8
Transmit buffer status register 12	RSCAN0TMSTS12	00_H	<RS-CAN0_base> + $02DC_H$	8
Transmit buffer status register 13	RSCAN0TMSTS13	00_H	<RS-CAN0_base> + $02DD_H$	8
Transmit buffer status register 14	RSCAN0TMSTS14	00_H	<RS-CAN0_base> + $02DE_H$	8
Transmit buffer status register 15	RSCAN0TMSTS15	00_H	<RS-CAN0_base> + $02DF_H$	8
Transmit buffer status register 16	RSCAN0TMSTS16	00_H	<RS-CAN0_base> + $02E0_H$	8
Transmit buffer status register 17	RSCAN0TMSTS17	00_H	<RS-CAN0_base> + $02E1_H$	8
Transmit buffer status register 18	RSCAN0TMSTS18	00_H	<RS-CAN0_base> + $02E2_H$	8
Transmit buffer status register 19	RSCAN0TMSTS19	00_H	<RS-CAN0_base> + $02E3_H$	8
Transmit buffer status register 20	RSCAN0TMSTS20	00_H	<RS-CAN0_base> + $02E4_H$	8
Transmit buffer status register 21	RSCAN0TMSTS21	00_H	<RS-CAN0_base> + $02E5_H$	8
Transmit buffer status register 22	RSCAN0TMSTS22	00_H	<RS-CAN0_base> + $02E6_H$	8
Transmit buffer status register 23	RSCAN0TMSTS23	00_H	<RS-CAN0_base> + $02E7_H$	8
Transmit buffer status register 24	RSCAN0TMSTS24	00_H	<RS-CAN0_base> + $02E8_H$	8
Transmit buffer status register 25	RSCAN0TMSTS25	00_H	<RS-CAN0_base> + $02E9_H$	8
Transmit buffer status register 26	RSCAN0TMSTS26	00_H	<RS-CAN0_base> + $02EA_H$	8
Transmit buffer status register 27	RSCAN0TMSTS27	00_H	<RS-CAN0_base> + $02EB_H$	8
Transmit buffer status register 28	RSCAN0TMSTS28	00_H	<RS-CAN0_base> + $02EC_H$	8
Transmit buffer status register 29	RSCAN0TMSTS29	00_H	<RS-CAN0_base> + $02ED_H$	8
Transmit buffer status register 30	RSCAN0TMSTS30	00_H	<RS-CAN0_base> + $02EE_H$	8
Transmit buffer status register 31	RSCAN0TMSTS31	00_H	<RS-CAN0_base> + $02EF_H$	8
Transmit buffer status register 32	RSCAN0TMSTS32	00_H	<RS-CAN0_base> + $02F0_H$	8

Table 16.10 List of RS-CAN Module Registers (5/21)

Register Name	Symbol	Value after Reset	Address	Access Size
Transmit buffer status register 33	RSCAN0TMSTS33	00_H	<RS-CAN0_base> + $02F1_H$	8
Transmit buffer status register 34	RSCAN0TMSTS34	00_H	<RS-CAN0_base> + $02F2_H$	8
Transmit buffer status register 35	RSCAN0TMSTS35	00_H	<RS-CAN0_base> + $02F3_H$	8
Transmit buffer status register 36	RSCAN0TMSTS36	00_H	<RS-CAN0_base> + $02F4_H$	8
Transmit buffer status register 37	RSCAN0TMSTS37	00_H	<RS-CAN0_base> + $02F5_H$	8
Transmit buffer status register 38	RSCAN0TMSTS38	00_H	<RS-CAN0_base> + $02F6_H$	8
Transmit buffer status register 39	RSCAN0TMSTS39	00_H	<RS-CAN0_base> + $02F7_H$	8
Transmit buffer status register 40	RSCAN0TMSTS40	00_H	<RS-CAN0_base> + $02F8_H$	8
Transmit buffer status register 41	RSCAN0TMSTS41	00_H	<RS-CAN0_base> + $02F9_H$	8
Transmit buffer status register 42	RSCAN0TMSTS42	00_H	<RS-CAN0_base> + $02FA_H$	8
Transmit buffer status register 43	RSCAN0TMSTS43	00_H	<RS-CAN0_base> + $02FB_H$	8
Transmit buffer status register 44	RSCAN0TMSTS44	00_H	<RS-CAN0_base> + $02FC_H$	8
Transmit buffer status register 45	RSCAN0TMSTS45	00_H	<RS-CAN0_base> + $02FD_H$	8
Transmit buffer status register 46	RSCAN0TMSTS46	00_H	<RS-CAN0_base> + $02FE_H$	8
Transmit buffer status register 47	RSCAN0TMSTS47	00_H	<RS-CAN0_base> + $02FF_H$	8
Transmit buffer status register 48	RSCAN0TMSTS48	00_H	<RS-CAN0_base> + 0300_H	8
Transmit buffer status register 49	RSCAN0TMSTS49	00_H	<RS-CAN0_base> + 0301_H	8
Transmit buffer status register 50	RSCAN0TMSTS50	00_H	<RS-CAN0_base> + 0302_H	8
Transmit buffer status register 51	RSCAN0TMSTS51	00_H	<RS-CAN0_base> + 0303_H	8
Transmit buffer status register 52	RSCAN0TMSTS52	00_H	<RS-CAN0_base> + 0304_H	8
Transmit buffer status register 53	RSCAN0TMSTS53	00_H	<RS-CAN0_base> + 0305_H	8
Transmit buffer status register 54	RSCAN0TMSTS54	00_H	<RS-CAN0_base> + 0306_H	8
Transmit buffer status register 55	RSCAN0TMSTS55	00_H	<RS-CAN0_base> + 0307_H	8
Transmit buffer status register 56	RSCAN0TMSTS56	00_H	<RS-CAN0_base> + 0308_H	8
Transmit buffer status register 57	RSCAN0TMSTS57	00_H	<RS-CAN0_base> + 0309_H	8
Transmit buffer status register 58	RSCAN0TMSTS58	00_H	<RS-CAN0_base> + $030A_H$	8
Transmit buffer status register 59	RSCAN0TMSTS59	00_H	<RS-CAN0_base> + $030B_H$	8
Transmit buffer status register 60	RSCAN0TMSTS60	00_H	<RS-CAN0_base> + $030C_H$	8
Transmit buffer status register 61	RSCAN0TMSTS61	00_H	<RS-CAN0_base> + $030D_H$	8
Transmit buffer status register 62	RSCAN0TMSTS62	00_H	<RS-CAN0_base> + $030E_H$	8
Transmit buffer status register 63	RSCAN0TMSTS63	00_H	<RS-CAN0_base> + $030F_H$	8
Transmit buffer transmit request status register 0	RSCAN0TMTRSTS0	$0000\ 0000_H$	<RS-CAN0_base> + 0350_H	8, 16, 32
Transmit buffer transmit request status register 1	RSCAN0TMTRSTS1	$0000\ 0000_H$	<RS-CAN0_base> + 0354_H	8, 16, 32
Transmit buffer transmit abort request status register 0	RSCAN0TMTARSTS0	$0000\ 0000_H$	<RS-CAN0_base> + 0360_H	8, 16, 32
Transmit buffer transmit abort request status register 1	RSCAN0TMTARSTS1	$0000\ 0000_H$	<RS-CAN0_base> + 0364_H	8, 16, 32
Transmit buffer transmit complete status register 0	RSCAN0TMTCSTS0	$0000\ 0000_H$	<RS-CAN0_base> + 0370_H	8, 16, 32
Transmit buffer transmit complete status register 1	RSCAN0TMTCSTS1	$0000\ 0000_H$	<RS-CAN0_base> + 0374_H	8, 16, 32
Transmit buffer transmit abort status register 0	RSCAN0TMTASTS0	$0000\ 0000_H$	<RS-CAN0_base> + 0380_H	8, 16, 32
Transmit buffer transmit abort status register 1	RSCAN0TMTASTS1	$0000\ 0000_H$	<RS-CAN0_base> + 0384_H	8, 16, 32
Transmit buffer interrupt enable configuration register 0	RSCAN0TMIEC0	$0000\ 0000_H$	<RS-CAN0_base> + 0390_H	8, 16, 32
Transmit buffer interrupt enable configuration register 1	RSCAN0TMIEC1	$0000\ 0000_H$	<RS-CAN0_base> + 0394_H	8, 16, 32
Transmit queue configuration/control register 0	RSCAN0TXQCC0	$0000\ 0000_H$	<RS-CAN0_base> + $03A0_H$	8, 16, 32
Transmit queue configuration/control register 1	RSCAN0TXQCC1	$0000\ 0000_H$	<RS-CAN0_base> + $03A4_H$	8, 16, 32
Transmit queue configuration/control register 2	RSCAN0TXQCC2	$0000\ 0000_H$	<RS-CAN0_base> + $03A8_H$	8, 16, 32
Transmit queue configuration/control register 3	RSCAN0TXQCC3	$0000\ 0000_H$	<RS-CAN0_base> + $03AC_H$	8, 16, 32
Transmit queue status register 0	RSCAN0TXQSTS0	$0000\ 0001_H$	<RS-CAN0_base> + $03C0_H$	8, 16, 32
Transmit queue status register 1	RSCAN0TXQSTS1	$0000\ 0001_H$	<RS-CAN0_base> + $03C4_H$	8, 16, 32
Transmit queue status register2	RSCAN0TXQSTS2	$0000\ 0001_H$	<RS-CAN0_base> + $03C8_H$	8, 16, 32
Transmit queue status register 3	RSCAN0TXQSTS3	$0000\ 0001_H$	<RS-CAN0_base> + $03CC_H$	8, 16, 32

Table 16.10 List of RS-CAN Module Registers (6/21)

Register Name	Symbol	Value after Reset	Address	Access Size
Transmit queue pointer control register 0	RSCAN0TXQPCTR0	0000 0000_H	<RS-CAN0_base> + $03E0_H$	8, 16, 32
Transmit queue pointer control register 1	RSCAN0TXQPCTR1	0000 0000_H	<RS-CAN0_base> + $03E4_H$	8, 16, 32
Transmit queue pointer control register 2	RSCAN0TXQPCTR2	0000 0000_H	<RS-CAN0_base> + $03E8_H$	8, 16, 32
Transmit queue pointer control register 3	RSCAN0TXQPCTR3	0000 0000_H	<RS-CAN0_base> + $03EC_H$	8, 16, 32
Transmit history list configuration/control register 0	RSCAN0THLCC0	0000 0000_H	<RS-CAN0_base> + 0400_H	8, 16, 32
Transmit history list configuration/control register 1	RSCAN0THLCC1	0000 0000_H	<RS-CAN0_base> + 0404_H	8, 16, 32
Transmit history list configuration/control register 2	RSCAN0THLCC2	0000 0000_H	<RS-CAN0_base> + 0408_H	8, 16, 32
Transmit history list configuration/control register 3	RSCAN0THLCC3	0000 0000_H	<RS-CAN0_base> + $040C_H$	8, 16, 32
Transmit history list status register 0	RSCAN0THLSTS0	0000 0001_H	<RS-CAN0_base> + 0420_H	8, 16, 32
Transmit history list status register 1	RSCAN0THLSTS1	0000 0001_H	<RS-CAN0_base> + 0424_H	8, 16, 32
Transmit history list status register 2	RSCAN0THLSTS2	0000 0001_H	<RS-CAN0_base> + 0428_H	8, 16, 32
Transmit history list status register 3	RSCAN0THLSTS3	0000 0001_H	<RS-CAN0_base> + $042C_H$	8, 16, 32
Transmit history list pointer control register 0	RSCAN0THLPCTR0	0000 0000_H	<RS-CAN0_base> + 0440_H	8, 16, 32
Transmit history list pointer control register 1	RSCAN0THLPCTR1	0000 0000_H	<RS-CAN0_base> + 0444_H	8, 16, 32
Transmit history list pointer control register 2	RSCAN0THLPCTR2	0000 0000_H	<RS-CAN0_base> + 0448_H	8, 16, 32
Transmit history list pointer control register 3	RSCAN0THLPCTR3	0000 0000_H	<RS-CAN0_base> + $044C_H$	8, 16, 32
Global TX interrupt status register 0	RSCAN0GTINTSTS0	0000 0000_H	<RS-CAN0_base> + 0460_H	8, 16, 32
Global test configuration register	RSCAN0GTSTCFG	0000 0000_H	<RS-CAN0_base> + 0468_H	8, 16, 32
Global test control register	RSCAN0GTSTCTR	0000 0000_H	<RS-CAN0_base> + $046C_H$	8, 16, 32
Global lock key register	RSCAN0GLOCKK	0000 0000_H	<RS-CAN0_base> + $047C_H$	16, 32
Global acceptance filter list ID register 0	RSCAN0GAFLID0	0000 0000_H*1	<RS-CAN0_base> + 0500_H	8, 16, 32
Global acceptance filter list mask register 0	RSCAN0GAFLM0	0000 0000_H*1	<RS-CAN0_base> + 0504_H	8, 16, 32
Global acceptance filter list pointer 0 register 0	RSCAN0GAFLP00	0000 0000_H*1	<RS-CAN0_base> + 0508_H	8, 16, 32
Global acceptance filter list pointer 1 register 0	RSCAN0GAFLP10	0000 0000_H*1	<RS-CAN0_base> + $050C_H$	8, 16, 32
Global acceptance filter list ID register 1	RSCAN0GAFLID1	0000 0000_H*1	<RS-CAN0_base> + 0510_H	8, 16, 32
Global acceptance filter list mask register 1	RSCAN0GAFLM1	0000 0000_H*1	<RS-CAN0_base> + 0514_H	8, 16, 32
Global acceptance filter list pointer 0 register 1	RSCAN0GAFLP01	0000 0000_H*1	<RS-CAN0_base> + 0518_H	8, 16, 32
Global acceptance filter list pointer 1 register 1	RSCAN0GAFLP11	0000 0000_H*1	<RS-CAN0_base> + $051C_H$	8, 16, 32
Global acceptance filter list ID register 2	RSCAN0GAFLID2	0000 0000_H*1	<RS-CAN0_base> + 0520_H	8, 16, 32
Global acceptance filter list mask register 2	RSCAN0GAFLM2	0000 0000_H*1	<RS-CAN0_base> + 0524_H	8, 16, 32
Global acceptance filter list pointer 0 register 2	RSCAN0GAFLP02	0000 0000_H*1	<RS-CAN0_base> + 0528_H	8, 16, 32
Global acceptance filter list pointer 1 register 2	RSCAN0GAFLP12	0000 0000_H*1	<RS-CAN0_base> + $052C_H$	8, 16, 32
Global acceptance filter list ID register 3	RSCAN0GAFLID3	0000 0000_H*1	<RS-CAN0_base> + 0530_H	8, 16, 32
Global acceptance filter list mask register 3	RSCAN0GAFLM3	0000 0000_H*1	<RS-CAN0_base> + 0534_H	8, 16, 32
Global acceptance filter list pointer 0 register 3	RSCAN0GAFLP03	0000 0000_H*1	<RS-CAN0_base> + 0538_H	8, 16, 32
Global acceptance filter list pointer 1 register 3	RSCAN0GAFLP13	0000 0000_H*1	<RS-CAN0_base> + $053C_H$	8, 16, 32
Global acceptance filter list ID register 4	RSCAN0GAFLID4	0000 0000_H*1	<RS-CAN0_base> + 0540_H	8, 16, 32
Global acceptance filter list mask register 4	RSCAN0GAFLM4	0000 0000_H*1	<RS-CAN0_base> + 0544_H	8, 16, 32
Global acceptance filter list pointer 0 register 4	RSCAN0GAFLP04	0000 0000_H*1	<RS-CAN0_base> + 0548_H	8, 16, 32
Global acceptance filter list pointer 1 register 4	RSCAN0GAFLP14	0000 0000_H*1	<RS-CAN0_base> + $054C_H$	8, 16, 32
Global acceptance filter list ID register 5	RSCAN0GAFLID5	0000 0000_H*1	<RS-CAN0_base> + 0550_H	8, 16, 32
Global acceptance filter list mask register 5	RSCAN0GAFLM5	0000 0000_H*1	<RS-CAN0_base> + 0554_H	8, 16, 32
Global acceptance filter list pointer 0 register 5	RSCAN0GAFLP05	0000 0000_H*1	<RS-CAN0_base> + 0558_H	8, 16, 32
Global acceptance filter list pointer 1 register 5	RSCAN0GAFLP15	0000 0000_H*1	<RS-CAN0_base> + $055C_H$	8, 16, 32
Global acceptance filter list ID register 6	RSCAN0GAFLID6	0000 0000_H*1	<RS-CAN0_base> + 0560_H	8, 16, 32
Global acceptance filter list mask register 6	RSCAN0GAFLM6	0000 0000_H*1	<RS-CAN0_base> + 0564_H	8, 16, 32
Global acceptance filter list pointer 0 register 6	RSCAN0GAFLP06	0000 0000_H*1	<RS-CAN0_base> + 0568_H	8, 16, 32
Global acceptance filter list pointer 1 register 6	RSCAN0GAFLP16	0000 0000_H*1	<RS-CAN0_base> + $056C_H$	8, 16, 32
Global acceptance filter list ID register 7	RSCAN0GAFLID7	0000 0000_H*1	<RS-CAN0_base> + 0570_H	8, 16, 32

Table 16.10 List of RS-CAN Module Registers (7/21)

Register Name	Symbol	Value after Reset	Address	Access Size
Global acceptance filter list mask register 7	RSCAN0GAFLM7	0000 0000_H*1	<RS-CAN0_base> + 0574_H	8, 16, 32
Global acceptance filter list pointer 0 register 7	RSCAN0GAFLP07	0000 0000_H*1	<RS-CAN0_base> + 0578_H	8, 16, 32
Global acceptance filter list pointer 1 register 7	RSCAN0GAFLP17	0000 0000_H*1	<RS-CAN0_base> + $057C_H$	8, 16, 32
Global acceptance filter list ID register 8	RSCAN0GAFLID8	0000 0000_H*1	<RS-CAN0_base> + 0580_H	8, 16, 32
Global acceptance filter list mask register 8	RSCAN0GAFLM8	0000 0000_H*1	<RS-CAN0_base> + 0584_H	8, 16, 32
Global acceptance filter list pointer 0 register 8	RSCAN0GAFLP08	0000 0000_H*1	<RS-CAN0_base> + 0588_H	8, 16, 32
Global acceptance filter list pointer 1 register 8	RSCAN0GAFLP18	0000 0000_H*1	<RS-CAN0_base> + $058C_H$	8, 16, 32
Global acceptance filter list ID register 9	RSCAN0GAFLID9	0000 0000_H*1	<RS-CAN0_base> + 0590_H	8, 16, 32
Global acceptance filter list mask register 9	RSCAN0GAFLM9	0000 0000_H*1	<RS-CAN0_base> + 0594_H	8, 16, 32
Global acceptance filter list pointer 0 register 9	RSCAN0GAFLP09	0000 0000_H*1	<RS-CAN0_base> + 0598_H	8, 16, 32
Global acceptance filter list pointer 1 register 9	RSCAN0GAFLP19	0000 0000_H*1	<RS-CAN0_base> + $059C_H$	8, 16, 32
Global acceptance filter list ID register 10	RSCAN0GAFLID10	0000 0000_H*1	<RS-CAN0_base> + $05A0_H$	8, 16, 32
Global acceptance filter list mask register 10	RSCAN0GAFLM10	0000 0000_H*1	<RS-CAN0_base> + $05A4_H$	8, 16, 32
Global acceptance filter list pointer 0 register 10	RSCAN0GAFLP010	0000 0000_H*1	<RS-CAN0_base> + $05A8_H$	8, 16, 32
Global acceptance filter list pointer 1 register 10	RSCAN0GAFLP110	0000 0000_H*1	<RS-CAN0_base> + $05AC_H$	8, 16, 32
Global acceptance filter list ID register 11	RSCAN0GAFLID11	0000 0000_H*1	<RS-CAN0_base> + $05B0_H$	8, 16, 32
Global acceptance filter list mask register 11	RSCAN0GAFLM11	0000 0000_H*1	<RS-CAN0_base> + $05B4_H$	8, 16, 32
Global acceptance filter list pointer 0 register 11	RSCAN0GAFLP011	0000 0000_H*1	<RS-CAN0_base> + $05B8_H$	8, 16, 32
Global acceptance filter list pointer 1 register 11	RSCAN0GAFLP111	0000 0000_H*1	<RS-CAN0_base> + $05BC_H$	8, 16, 32
Global acceptance filter list ID register 12	RSCAN0GAFLID12	0000 0000_H*1	<RS-CAN0_base> + $05C0_H$	8, 16, 32
Global acceptance filter list mask register 12	RSCAN0GAFLM12	0000 0000_H*1	<RS-CAN0_base> + $05C4_H$	8, 16, 32
Global acceptance filter list pointer 0 register 12	RSCAN0GAFLP012	0000 0000_H*1	<RS-CAN0_base> + $05C8_H$	8, 16, 32
Global acceptance filter list pointer 1 register 12	RSCAN0GAFLP112	0000 0000_H*1	<RS-CAN0_base> + $05CC_H$	8, 16, 32
Global acceptance filter list ID register 13	RSCAN0GAFLID13	0000 0000_H*1	<RS-CAN0_base> + $05D0_H$	8, 16, 32
Global acceptance filter list mask register 13	RSCAN0GAFLM13	0000 0000_H*1	<RS-CAN0_base> + $05D4_H$	8, 16, 32
Global acceptance filter list pointer 0 register 13	RSCAN0GAFLP013	0000 0000_H*1	<RS-CAN0_base> + $05D8_H$	8, 16, 32
Global acceptance filter list pointer 1 register 13	RSCAN0GAFLP113	0000 0000_H*1	<RS-CAN0_base> + $05DC_H$	8, 16, 32
Global acceptance filter list ID register 14	RSCAN0GAFLID14	0000 0000_H*1	<RS-CAN0_base> + $05E0_H$	8, 16, 32
Global acceptance filter list mask register 14	RSCAN0GAFLM14	0000 0000_H*1	<RS-CAN0_base> + $05E4_H$	8, 16, 32
Global acceptance filter list pointer 0 register 14	RSCAN0GAFLP014	0000 0000_H*1	<RS-CAN0_base> + $05E8_H$	8, 16, 32
Global acceptance filter list pointer 1 register 14	RSCAN0GAFLP114	0000 0000_H*1	<RS-CAN0_base> + $05EC_H$	8, 16, 32
Global acceptance filter list ID register 15	RSCAN0GAFLID15	0000 0000_H*1	<RS-CAN0_base> + $05F0_H$	8, 16, 32
Global acceptance filter list mask register 15	RSCAN0GAFLM15	0000 0000_H*1	<RS-CAN0_base> + $05F4_H$	8, 16, 32
Global acceptance filter list pointer 0 register 15	RSCAN0GAFLP015	0000 0000_H*1	<RS-CAN0_base> + $05F8_H$	8, 16, 32
Global acceptance filter list pointer 1 register 15	RSCAN0GAFLP115	0000 0000_H*1	<RS-CAN0_base> + $05FC_H$	8, 16, 32
Receive buffer ID register 0	RSCAN0RMID0	0000 0000_H*1	<RS-CAN0_base> + 0600_H	8, 16, 32
Receive buffer pointer register 0	RSCAN0RMPTR0	0000 0000_H*1	<RS-CAN0_base> + 0604_H	8, 16, 32
Receive buffer data field 0 register 0	RSCAN0RMDF00	0000 0000_H*1	<RS-CAN0_base> + 0608_H	8, 16, 32
Receive buffer data field 1 register 0	RSCAN0RMDF10	0000 0000_H*1	<RS-CAN0_base> + $060C_H$	8, 16, 32
Receive buffer ID register 1	RSCAN0RMID1	0000 0000_H*1	<RS-CAN0_base> + 0610_H	8, 16, 32
Receive buffer pointer register 1	RSCAN0RMPTR1	0000 0000_H*1	<RS-CAN0_base> + 0614_H	8, 16, 32
Receive buffer data field 0 register 1	RSCAN0RMDF01	0000 0000_H*1	<RS-CAN0_base> + 0618_H	8, 16, 32
Receive buffer data field 1 register 1	RSCAN0RMDF11	0000 0000_H*1	<RS-CAN0_base> + $061C_H$	8, 16, 32
Receive buffer ID register 2	RSCAN0RMID2	0000 0000_H*1	<RS-CAN0_base> + 0620_H	8, 16, 32
Receive buffer pointer register 2	RSCAN0RMPTR2	0000 0000_H*1	<RS-CAN0_base> + 0624_H	8, 16, 32
Receive buffer data field 0 register 2	RSCAN0RMDF02	0000 0000_H*1	<RS-CAN0_base> + 0628_H	8, 16, 32
Receive buffer data field 1 register 2	RSCAN0RMDF12	0000 0000_H*1	<RS-CAN0_base> + $062C_H$	8, 16, 32
Receive buffer ID register 3	RSCAN0RMID3	0000 0000_H*1	<RS-CAN0_base> + 0630_H	8, 16, 32
Receive buffer pointer register 3	RSCAN0RMPTR3	0000 0000_H*1	<RS-CAN0_base> + 0634_H	8, 16, 32

Table 16.10 List of RS-CAN Module Registers (8/21)

Register Name	Symbol	Value after Reset	Address	Access Size
Receive buffer data field 0 register 3	RSCAN0RMDF03	0000 0000_H*1	<RS-CAN0_base> + 0638_H	8, 16, 32
Receive buffer data field 1 register 3	RSCAN0RMDF13	0000 0000_H*1	<RS-CAN0_base> + $063C_H$	8, 16, 32
Receive buffer ID register 4	RSCAN0RMID4	0000 0000_H*1	<RS-CAN0_base> + 0640_H	8, 16, 32
Receive buffer pointer register 4	RSCAN0RMPTR4	0000 0000_H*1	<RS-CAN0_base> + 0644_H	8, 16, 32
Receive buffer data field 0 register 4	RSCAN0RMDF04	0000 0000_H*1	<RS-CAN0_base> + 0648_H	8, 16, 32
Receive buffer data field 1 register 4	RSCAN0RMDF14	0000 0000_H*1	<RS-CAN0_base> + $064C_H$	8, 16, 32
Receive buffer ID register 5	RSCAN0RMID5	0000 0000_H*1	<RS-CAN0_base> + 0650_H	8, 16, 32
Receive buffer pointer register 5	RSCAN0RMPTR5	0000 0000_H*1	<RS-CAN0_base> + 0654_H	8, 16, 32
Receive buffer data field 0 register 5	RSCAN0RMDF05	0000 0000_H*1	<RS-CAN0_base> + 0658_H	8, 16, 32
Receive buffer data field 1 register 5	RSCAN0RMDF15	0000 0000_H*1	<RS-CAN0_base> + $065C_H$	8, 16, 32
Receive buffer ID register 6	RSCAN0RMID6	0000 0000_H*1	<RS-CAN0_base> + 0660_H	8, 16, 32
Receive buffer pointer register 6	RSCAN0RMPTR6	0000 0000_H*1	<RS-CAN0_base> + 0664_H	8, 16, 32
Receive buffer data field 0 register 6	RSCAN0RMDF06	0000 0000_H*1	<RS-CAN0_base> + 0668_H	8, 16, 32
Receive buffer data field 1 register 6	RSCAN0RMDF16	0000 0000_H*1	<RS-CAN0_base> + $066C_H$	8, 16, 32
Receive buffer ID register 7	RSCAN0RMID7	0000 0000_H*1	<RS-CAN0_base> + 0670_H	8, 16, 32
Receive buffer pointer register 7	RSCAN0RMPTR7	0000 0000_H*1	<RS-CAN0_base> + 0674_H	8, 16, 32
Receive buffer data field 0 register 7	RSCAN0RMDF07	0000 0000_H*1	<RS-CAN0_base> + 0678_H	8, 16, 32
Receive buffer data field 1 register 7	RSCAN0RMDF17	0000 0000_H*1	<RS-CAN0_base> + $067C_H$	8, 16, 32
Receive buffer ID register 8	RSCAN0RMID8	0000 0000_H*1	<RS-CAN0_base> + 0680_H	8, 16, 32
Receive buffer pointer register 8	RSCAN0RMPTR8	0000 0000_H*1	<RS-CAN0_base> + 0684_H	8, 16, 32
Receive buffer data field 0 register 8	RSCAN0RMDF08	0000 0000_H*1	<RS-CAN0_base> + 0688_H	8, 16, 32
Receive buffer data field 1 register 8	RSCAN0RMDF18	0000 0000_H*1	<RS-CAN0_base> + $068C_H$	8, 16, 32
Receive buffer ID register 9	RSCAN0RMID9	0000 0000_H*1	<RS-CAN0_base> + 0690_H	8, 16, 32
Receive buffer pointer register 9	RSCAN0RMPTR9	0000 0000_H*1	<RS-CAN0_base> + 0694_H	8, 16, 32
Receive buffer data field 0 register 9	RSCAN0RMDF09	0000 0000_H*1	<RS-CAN0_base> + 0698_H	8, 16, 32
Receive buffer data field 1 register 9	RSCAN0RMDF19	0000 0000_H*1	<RS-CAN0_base> + $069C_H$	8, 16, 32
Receive buffer ID register 10	RSCAN0RMID10	0000 0000_H*1	<RS-CAN0_base> + $06A0_H$	8, 16, 32
Receive buffer pointer register 10	RSCAN0RMPTR10	0000 0000_H*1	<RS-CAN0_base> + $06A4_H$	8, 16, 32
Receive buffer data field 0 register 10	RSCAN0RMDF010	0000 0000_H*1	<RS-CAN0_base> + $06A8_H$	8, 16, 32
Receive buffer data field 1 register 10	RSCAN0RMDF110	0000 0000_H*1	<RS-CAN0_base> + $06AC_H$	8, 16, 32
Receive buffer ID register 11	RSCAN0RMID11	0000 0000_H*1	<RS-CAN0_base> + $06B0_H$	8, 16, 32
Receive buffer pointer register 11	RSCAN0RMPTR11	0000 0000_H*1	<RS-CAN0_base> + $06B4_H$	8, 16, 32
Receive buffer data field 0 register 11	RSCAN0RMDF011	0000 0000_H*1	<RS-CAN0_base> + $06B8_H$	8, 16, 32
Receive buffer data field 1 register 11	RSCAN0RMDF111	0000 0000_H*1	<RS-CAN0_base> + $06BC_H$	8, 16, 32
Receive buffer ID register 12	RSCAN0RMID12	0000 0000_H*1	<RS-CAN0_base> + $06C0_H$	8, 16, 32
Receive buffer pointer register 12	RSCAN0RMPTR12	0000 0000_H*1	<RS-CAN0_base> + $06C4_H$	8, 16, 32
Receive buffer data field 0 register 12	RSCAN0RMDF012	0000 0000_H*1	<RS-CAN0_base> + $06C8_H$	8, 16, 32
Receive buffer data field 1 register 12	RSCAN0RMDF112	0000 0000_H*1	<RS-CAN0_base> + $06CC_H$	8, 16, 32
Receive buffer ID register 13	RSCAN0RMID13	0000 0000_H*1	<RS-CAN0_base> + $06D0_H$	8, 16, 32
Receive buffer pointer register 13	RSCAN0RMPTR13	0000 0000_H*1	<RS-CAN0_base> + $06D4_H$	8, 16, 32
Receive buffer data field 0 register 13	RSCAN0RMDF013	0000 0000_H*1	<RS-CAN0_base> + $06D8_H$	8, 16, 32
Receive buffer data field 1 register 13	RSCAN0RMDF113	0000 0000_H*1	<RS-CAN0_base> + $06DC_H$	8, 16, 32
Receive buffer ID register 14	RSCAN0RMID14	0000 0000_H*1	<RS-CAN0_base> + $06E0_H$	8, 16, 32
Receive buffer pointer register 14	RSCAN0RMPTR14	0000 0000_H*1	<RS-CAN0_base> + $06E4_H$	8, 16, 32
Receive buffer data field 0 register 14	RSCAN0RMDF014	0000 0000_H*1	<RS-CAN0_base> + $06E8_H$	8, 16, 32
Receive buffer data field 1 register 14	RSCAN0RMDF114	0000 0000_H*1	<RS-CAN0_base> + $06EC_H$	8, 16, 32
Receive buffer ID register 15	RSCAN0RMID15	0000 0000_H*1	<RS-CAN0_base> + $06F0_H$	8, 16, 32
Receive buffer pointer register 15	RSCAN0RMPTR15	0000 0000_H*1	<RS-CAN0_base> + $06F4_H$	8, 16, 32
Receive buffer data field 0 register 15	RSCAN0RMDF015	0000 0000_H*1	<RS-CAN0_base> + $06F8_H$	8, 16, 32

Table 16.10 List of RS-CAN Module Registers (9/21)

Register Name	Symbol	Value after Reset	Address	Access Size
Receive buffer data field 1 register 15	RSCAN0RMDF115	0000 0000_H*1	<RS-CAN0_base> + $06FC_H$	8, 16, 32
Receive buffer ID register 16	RSCAN0RMID16	0000 0000_H*1	<RS-CAN0_base> + 0700_H	8, 16, 32
Receive buffer pointer register 16	RSCAN0RMPTR16	0000 0000_H*1	<RS-CAN0_base> + 0704_H	8, 16, 32
Receive buffer data field 0 register 16	RSCAN0RMDF016	0000 0000_H*1	<RS-CAN0_base> + 0708_H	8, 16, 32
Receive buffer data field 1 register 16	RSCAN0RMDF116	0000 0000_H*1	<RS-CAN0_base> + $070C_H$	8, 16, 32
Receive buffer ID register 17	RSCAN0RMID17	0000 0000_H*1	<RS-CAN0_base> + 0710_H	8, 16, 32
Receive buffer pointer register 17	RSCAN0RMPTR17	0000 0000_H*1	<RS-CAN0_base> + 0714_H	8, 16, 32
Receive buffer data field 0 register 17	RSCAN0RMDF017	0000 0000_H*1	<RS-CAN0_base> + 0718_H	8, 16, 32
Receive buffer data field 1 register 17	RSCAN0RMDF117	0000 0000_H*1	<RS-CAN0_base> + $071C_H$	8, 16, 32
Receive buffer ID register 18	RSCAN0RMID18	0000 0000_H*1	<RS-CAN0_base> + 0720_H	8, 16, 32
Receive buffer pointer register 18	RSCAN0RMPTR18	0000 0000_H*1	<RS-CAN0_base> + 0724_H	8, 16, 32
Receive buffer data field 0 register 18	RSCAN0RMDF018	0000 0000_H*1	<RS-CAN0_base> + 0728_H	8, 16, 32
Receive buffer data field 1 register 18	RSCAN0RMDF118	0000 0000_H*1	<RS-CAN0_base> + $072C_H$	8, 16, 32
Receive buffer ID register 19	RSCAN0RMID19	0000 0000_H*1	<RS-CAN0_base> + 0730_H	8, 16, 32
Receive buffer pointer register 19	RSCAN0RMPTR19	0000 0000_H*1	<RS-CAN0_base> + 0734_H	8, 16, 32
Receive buffer data field 0 register 19	RSCAN0RMDF019	0000 0000_H*1	<RS-CAN0_base> + 0738_H	8, 16, 32
Receive buffer data field 1 register 19	RSCAN0RMDF119	0000 0000_H*1	<RS-CAN0_base> + $073C_H$	8, 16, 32
Receive buffer ID register 20	RSCAN0RMID20	0000 0000_H*1	<RS-CAN0_base> + 0740_H	8, 16, 32
Receive buffer pointer register 20	RSCAN0RMPTR20	0000 0000_H*1	<RS-CAN0_base> + 0744_H	8, 16, 32
Receive buffer data field 0 register 20	RSCAN0RMDF020	0000 0000_H*1	<RS-CAN0_base> + 0748_H	8, 16, 32
Receive buffer data field 1 register 20	RSCAN0RMDF120	0000 0000_H*1	<RS-CAN0_base> + $074C_H$	8, 16, 32
Receive buffer ID register 21	RSCAN0RMID21	0000 0000_H*1	<RS-CAN0_base> + 0750_H	8, 16, 32
Receive buffer pointer register 21	RSCAN0RMPTR21	0000 0000_H*1	<RS-CAN0_base> + 0754_H	8, 16, 32
Receive buffer data field 0 register 21	RSCAN0RMDF021	0000 0000_H*1	<RS-CAN0_base> + 0758_H	8, 16, 32
Receive buffer data field 1 register 21	RSCAN0RMDF121	0000 0000_H*1	<RS-CAN0_base> + $075C_H$	8, 16, 32
Receive buffer ID register 22	RSCAN0RMID22	0000 0000_H*1	<RS-CAN0_base> + 0760_H	8, 16, 32
Receive buffer pointer register 22	RSCAN0RMPTR22	0000 0000_H*1	<RS-CAN0_base> + 0764_H	8, 16, 32
Receive buffer data field 0 register 22	RSCAN0RMDF022	0000 0000_H*1	<RS-CAN0_base> + 0768_H	8, 16, 32
Receive buffer data field 1 register 22	RSCAN0RMDF122	0000 0000_H*1	<RS-CAN0_base> + $076C_H$	8, 16, 32
Receive buffer ID register 23	RSCAN0RMID23	0000 0000_H*1	<RS-CAN0_base> + 0770_H	8, 16, 32
Receive buffer pointer register 23	RSCAN0RMPTR23	0000 0000_H*1	<RS-CAN0_base> + 0774_H	8, 16, 32
Receive buffer data field 0 register 23	RSCAN0RMDF023	0000 0000_H*1	<RS-CAN0_base> + 0778_H	8, 16, 32
Receive buffer data field 1 register 23	RSCAN0RMDF123	0000 0000_H*1	<RS-CAN0_base> + $077C_H$	8, 16, 32
Receive buffer ID register 24	RSCAN0RMID24	0000 0000_H*1	<RS-CAN0_base> + 0780_H	8, 16, 32
Receive buffer pointer register 24	RSCAN0RMPTR24	0000 0000_H*1	<RS-CAN0_base> + 0784_H	8, 16, 32
Receive buffer data field 0 register 24	RSCAN0RMDF024	0000 0000_H*1	<RS-CAN0_base> + 0788_H	8, 16, 32
Receive buffer data field 1 register 24	RSCAN0RMDF124	0000 0000_H*1	<RS-CAN0_base> + $078C_H$	8, 16, 32
Receive buffer ID register 25	RSCAN0RMID25	0000 0000_H*1	<RS-CAN0_base> + 0790_H	8, 16, 32
Receive buffer pointer register 25	RSCAN0RMPTR25	0000 0000_H*1	<RS-CAN0_base> + 0794_H	8, 16, 32
Receive buffer data field 0 register 25	RSCAN0RMDF025	0000 0000_H*1	<RS-CAN0_base> + 0798_H	8, 16, 32
Receive buffer data field 1 register 25	RSCAN0RMDF125	0000 0000_H*1	<RS-CAN0_base> + $079C_H$	8, 16, 32
Receive buffer ID register 26	RSCAN0RMID26	0000 0000_H*1	<RS-CAN0_base> + $07A0_H$	8, 16, 32
Receive buffer pointer register 26	RSCAN0RMPTR26	0000 0000_H*1	<RS-CAN0_base> + $07A4_H$	8, 16, 32
Receive buffer data field 0 register 26	RSCAN0RMDF026	0000 0000_H*1	<RS-CAN0_base> + $07A8_H$	8, 16, 32
Receive buffer data field 1 register 26	RSCAN0RMDF126	0000 0000_H*1	<RS-CAN0_base> + $07AC_H$	8, 16, 32
Receive buffer ID register 27	RSCAN0RMID27	0000 0000_H*1	<RS-CAN0_base> + $07B0_H$	8, 16, 32
Receive buffer pointer register 27	RSCAN0RMPTR27	0000 0000_H*1	<RS-CAN0_base> + $07B4_H$	8, 16, 32
Receive buffer data field 0 register 27	RSCAN0RMDF027	0000 0000_H*1	<RS-CAN0_base> + $07B8_H$	8, 16, 32
Receive buffer data field 1 register 27	RSCAN0RMDF127	0000 0000_H*1	<RS-CAN0_base> + $07BC_H$	8, 16, 32

Table 16.10 List of RS-CAN Module Registers (10/21)

Register Name	Symbol	Value after Reset	Address	Access Size
Receive buffer ID register 28	RSCAN0RMID28	0000 0000_H*1	<RS-CAN0_base> + $07C0_H$	8, 16, 32
Receive buffer pointer register 28	RSCAN0RMPTR28	0000 0000_H*1	<RS-CAN0_base> + $07C4_H$	8, 16, 32
Receive buffer data field 0 register 28	RSCAN0RMDF028	0000 0000_H*1	<RS-CAN0_base> + $07C8_H$	8, 16, 32
Receive buffer data field 1 register 28	RSCAN0RMDF128	0000 0000_H*1	<RS-CAN0_base> + $07CC_H$	8, 16, 32
Receive buffer ID register 29	RSCAN0RMID29	0000 0000_H*1	<RS-CAN0_base> + $07D0_H$	8, 16, 32
Receive buffer pointer register 29	RSCAN0RMPTR29	0000 0000_H*1	<RS-CAN0_base> + $07D4_H$	8, 16, 32
Receive buffer data field 0 register 29	RSCAN0RMDF029	0000 0000_H*1	<RS-CAN0_base> + $07D8_H$	8, 16, 32
Receive buffer data field 1 register 29	RSCAN0RMDF129	0000 0000_H*1	<RS-CAN0_base> + $07DC_H$	8, 16, 32
Receive buffer ID register 30	RSCAN0RMID30	0000 0000_H*1	<RS-CAN0_base> + $07E0_H$	8, 16, 32
Receive buffer pointer register 30	RSCAN0RMPTR30	0000 0000_H*1	<RS-CAN0_base> + $07E4_H$	8, 16, 32
Receive buffer data field 0 register 30	RSCAN0RMDF030	0000 0000_H*1	<RS-CAN0_base> + $07E8_H$	8, 16, 32
Receive buffer data field 1 register 30	RSCAN0RMDF130	0000 0000_H*1	<RS-CAN0_base> + $07EC_H$	8, 16, 32
Receive buffer ID register 31	RSCAN0RMID31	0000 0000_H*1	<RS-CAN0_base> + $07F0_H$	8, 16, 32
Receive buffer pointer register 31	RSCAN0RMPTR31	0000 0000_H*1	<RS-CAN0_base> + $07F4_H$	8, 16, 32
Receive buffer data field 0 register 31	RSCAN0RMDF031	0000 0000_H*1	<RS-CAN0_base> + $07F8_H$	8, 16, 32
Receive buffer data field 1 register 31	RSCAN0RMDF131	0000 0000_H*1	<RS-CAN0_base> + $07FC_H$	8, 16, 32
Receive buffer ID register 32	RSCAN0RMID32	0000 0000_H*1	<RS-CAN0_base> + 0800_H	8, 16, 32
Receive buffer pointer register 32	RSCAN0RMPTR32	0000 0000_H*1	<RS-CAN0_base> + 0804_H	8, 16, 32
Receive buffer data field 0 register 32	RSCAN0RMDF032	0000 0000_H*1	<RS-CAN0_base> + 0808_H	8, 16, 32
Receive buffer data field 1 register 32	RSCAN0RMDF132	0000 0000_H*1	<RS-CAN0_base> + $080C_H$	8, 16, 32
Receive buffer ID register 33	RSCAN0RMID33	0000 0000_H*1	<RS-CAN0_base> + 0810_H	8, 16, 32
Receive buffer pointer register 33	RSCAN0RMPTR33	0000 0000_H*1	<RS-CAN0_base> + 0814_H	8, 16, 32
Receive buffer data field 0 register 33	RSCAN0RMDF033	0000 0000_H*1	<RS-CAN0_base> + 0818_H	8, 16, 32
Receive buffer data field 1 register 33	RSCAN0RMDF133	0000 0000_H*1	<RS-CAN0_base> + $081C_H$	8, 16, 32
Receive buffer ID register 34	RSCAN0RMID34	0000 0000_H*1	<RS-CAN0_base> + 0820_H	8, 16, 32
Receive buffer pointer register 34	RSCAN0RMPTR34	0000 0000_H*1	<RS-CAN0_base> + 0824_H	8, 16, 32
Receive buffer data field 0 register 34	RSCAN0RMDF034	0000 0000_H*1	<RS-CAN0_base> + 0828_H	8, 16, 32
Receive buffer data field 1 register 34	RSCAN0RMDF134	0000 0000_H*1	<RS-CAN0_base> + $082C_H$	8, 16, 32
Receive buffer ID register 35	RSCAN0RMID35	0000 0000_H*1	<RS-CAN0_base> + 0830_H	8, 16, 32
Receive buffer pointer register 35	RSCAN0RMPTR35	0000 0000_H*1	<RS-CAN0_base> + 0834_H	8, 16, 32
Receive buffer data field 0 register 35	RSCAN0RMDF035	0000 0000_H*1	<RS-CAN0_base> + 0838_H	8, 16, 32
Receive buffer data field 1 register 35	RSCAN0RMDF135	0000 0000_H*1	<RS-CAN0_base> + $083C_H$	8, 16, 32
Receive buffer ID register 36	RSCAN0RMID36	0000 0000_H*1	<RS-CAN0_base> + 0840_H	8, 16, 32
Receive buffer pointer register 36	RSCAN0RMPTR36	0000 0000_H*1	<RS-CAN0_base> + 0844_H	8, 16, 32
Receive buffer data field 0 register 36	RSCAN0RMDF036	0000 0000_H*1	<RS-CAN0_base> + 0848_H	8, 16, 32
Receive buffer data field 1 register 36	RSCAN0RMDF136	0000 0000_H*1	<RS-CAN0_base> + $084C_H$	8, 16, 32
Receive buffer ID register 37	RSCAN0RMID37	0000 0000_H*1	<RS-CAN0_base> + 0850_H	8, 16, 32
Receive buffer pointer register 37	RSCAN0RMPTR37	0000 0000_H*1	<RS-CAN0_base> + 0854_H	8, 16, 32
Receive buffer data field 0 register 37	RSCAN0RMDF037	0000 0000_H*1	<RS-CAN0_base> + 0858_H	8, 16, 32
Receive buffer data field 1 register 37	RSCAN0RMDF137	0000 0000_H*1	<RS-CAN0_base> + $085C_H$	8, 16, 32
Receive buffer ID register 38	RSCAN0RMID38	0000 0000_H*1	<RS-CAN0_base> + 0860_H	8, 16, 32
Receive buffer pointer register 38	RSCAN0RMPTR38	0000 0000_H*1	<RS-CAN0_base> + 0864_H	8, 16, 32
Receive buffer data field 0 register 38	RSCAN0RMDF038	0000 0000_H*1	<RS-CAN0_base> + 0868_H	8, 16, 32
Receive buffer data field 1 register 38	RSCAN0RMDF138	0000 0000_H*1	<RS-CAN0_base> + $086C_H$	8, 16, 32
Receive buffer ID register 39	RSCAN0RMID39	0000 0000_H*1	<RS-CAN0_base> + 0870_H	8, 16, 32
Receive buffer pointer register 39	RSCAN0RMPTR39	0000 0000_H*1	<RS-CAN0_base> + 0874_H	8, 16, 32
Receive buffer data field 0 register 39	RSCAN0RMDF039	0000 0000_H*1	<RS-CAN0_base> + 0878_H	8, 16, 32
Receive buffer data field 1 register 39	RSCAN0RMDF139	0000 0000_H*1	<RS-CAN0_base> + $087C_H$	8, 16, 32
Receive buffer ID register 40	RSCAN0RMID40	0000 0000_H*1	<RS-CAN0_base> + 0880_H	8, 16, 32

Table 16.10 List of RS-CAN Module Registers (11/21)

Register Name	Symbol	Value after Reset	Address	Access Size
Receive buffer pointer register 40	RSCAN0RMPTR40	0000 0000_H*1	<RS-CAN0_base> + 0884_H	8, 16, 32
Receive buffer data field 0 register 40	RSCAN0RMDF040	0000 0000_H*1	<RS-CAN0_base> + 0888_H	8, 16, 32
Receive buffer data field 1 register 40	RSCAN0RMDF140	0000 0000_H*1	<RS-CAN0_base> + $088C_H$	8, 16, 32
Receive buffer ID register 41	RSCAN0RMID41	0000 0000_H*1	<RS-CAN0_base> + 0890_H	8, 16, 32
Receive buffer pointer register 41	RSCAN0RMPTR41	0000 0000_H*1	<RS-CAN0_base> + 0894_H	8, 16, 32
Receive buffer data field 0 register 41	RSCAN0RMDF041	0000 0000_H*1	<RS-CAN0_base> + 0898_H	8, 16, 32
Receive buffer data field 1 register 41	RSCAN0RMDF141	0000 0000_H*1	<RS-CAN0_base> + $089C_H$	8, 16, 32
Receive buffer ID register 42	RSCAN0RMID42	0000 0000_H*1	<RS-CAN0_base> + $08A0_H$	8, 16, 32
Receive buffer pointer register 42	RSCAN0RMPTR42	0000 0000_H*1	<RS-CAN0_base> + $08A4_H$	8, 16, 32
Receive buffer data field 0 register 42	RSCAN0RMDF042	0000 0000_H*1	<RS-CAN0_base> + $08A8_H$	8, 16, 32
Receive buffer data field 1 register 42	RSCAN0RMDF142	0000 0000_H*1	<RS-CAN0_base> + $08AC_H$	8, 16, 32
Receive buffer ID register 43	RSCAN0RMID43	0000 0000_H*1	<RS-CAN0_base> + $08B0_H$	8, 16, 32
Receive buffer pointer register 43	RSCAN0RMPTR43	0000 0000_H*1	<RS-CAN0_base> + $08B4_H$	8, 16, 32
Receive buffer data field 0 register 43	RSCAN0RMDF043	0000 0000_H*1	<RS-CAN0_base> + $08B8_H$	8, 16, 32
Receive buffer data field 1 register 43	RSCAN0RMDF143	0000 0000_H*1	<RS-CAN0_base> + $08BC_H$	8, 16, 32
Receive buffer ID register 44	RSCAN0RMID44	0000 0000_H*1	<RS-CAN0_base> + $08C0_H$	8, 16, 32
Receive buffer pointer register 44	RSCAN0RMPTR44	0000 0000_H*1	<RS-CAN0_base> + $08C4_H$	8, 16, 32
Receive buffer data field 0 register 44	RSCAN0RMDF044	0000 0000_H*1	<RS-CAN0_base> + $08C8_H$	8, 16, 32
Receive buffer data field 1 register 44	RSCAN0RMDF144	0000 0000_H*1	<RS-CAN0_base> + $08CC_H$	8, 16, 32
Receive buffer ID register 45	RSCAN0RMID45	0000 0000_H*1	<RS-CAN0_base> + $08D0_H$	8, 16, 32
Receive buffer pointer register 45	RSCAN0RMPTR45	0000 0000_H*1	<RS-CAN0_base> + $08D4_H$	8, 16, 32
Receive buffer data field 0 register 45	RSCAN0RMDF045	0000 0000_H*1	<RS-CAN0_base> + $08D8_H$	8, 16, 32
Receive buffer data field 1 register 45	RSCAN0RMDF145	0000 0000_H*1	<RS-CAN0_base> + $08DC_H$	8, 16, 32
Receive buffer ID register 46	RSCAN0RMID46	0000 0000_H*1	<RS-CAN0_base> + $08E0_H$	8, 16, 32
Receive buffer pointer register 46	RSCAN0RMPTR46	0000 0000_H*1	<RS-CAN0_base> + $08E4_H$	8, 16, 32
Receive buffer data field 0 register 46	RSCAN0RMDF046	0000 0000_H*1	<RS-CAN0_base> + $08E8_H$	8, 16, 32
Receive buffer data field 1 register 46	RSCAN0RMDF146	0000 0000_H*1	<RS-CAN0_base> + $08EC_H$	8, 16, 32
Receive buffer ID register 47	RSCAN0RMID47	0000 0000_H*1	<RS-CAN0_base> + $08F0_H$	8, 16, 32
Receive buffer pointer register 47	RSCAN0RMPTR47	0000 0000_H*1	<RS-CAN0_base> + $08F4_H$	8, 16, 32
Receive buffer data field 0 register 47	RSCAN0RMDF047	0000 0000_H*1	<RS-CAN0_base> + $08F8_H$	8, 16, 32
Receive buffer data field 1 register 47	RSCAN0RMDF147	0000 0000_H*1	<RS-CAN0_base> + $08FC_H$	8, 16, 32
Receive buffer ID register 48	RSCAN0RMID48	0000 0000_H*1	<RS-CAN0_base> + 0900_H	8, 16, 32
Receive buffer pointer register 48	RSCAN0RMPTR48	0000 0000_H*1	<RS-CAN0_base> + 0904_H	8, 16, 32
Receive buffer data field 0 register 48	RSCAN0RMDF048	0000 0000_H*1	<RS-CAN0_base> + 0908_H	8, 16, 32
Receive buffer data field 1 register 48	RSCAN0RMDF148	0000 0000_H*1	<RS-CAN0_base> + $090C_H$	8, 16, 32
Receive buffer ID register 49	RSCAN0RMID49	0000 0000_H*1	<RS-CAN0_base> + 0910_H	8, 16, 32
Receive buffer pointer register 49	RSCAN0RMPTR49	0000 0000_H*1	<RS-CAN0_base> + 0914_H	8, 16, 32
Receive buffer data field 0 register 49	RSCAN0RMDF049	0000 0000_H*1	<RS-CAN0_base> + 0918_H	8, 16, 32
Receive buffer data field 1 register 49	RSCAN0RMDF149	0000 0000_H*1	<RS-CAN0_base> + $091C_H$	8, 16, 32
Receive buffer ID register 50	RSCAN0RMID50	0000 0000_H*1	<RS-CAN0_base> + 0920_H	8, 16, 32
Receive buffer pointer register 50	RSCAN0RMPTR50	0000 0000_H*1	<RS-CAN0_base> + 0924_H	8, 16, 32
Receive buffer data field 0 register 50	RSCAN0RMDF050	0000 0000_H*1	<RS-CAN0_base> + 0928_H	8, 16, 32
Receive buffer data field 1 register 50	RSCAN0RMDF150	0000 0000_H*1	<RS-CAN0_base> + $092C_H$	8, 16, 32
Receive buffer ID register 51	RSCAN0RMID51	0000 0000_H*1	<RS-CAN0_base> + 0930_H	8, 16, 32
Receive buffer pointer register 51	RSCAN0RMPTR51	0000 0000_H*1	<RS-CAN0_base> + 0934_H	8, 16, 32
Receive buffer data field 0 register 51	RSCAN0RMDF051	0000 0000_H*1	<RS-CAN0_base> + 0938_H	8, 16, 32
Receive buffer data field 1 register 51	RSCAN0RMDF151	0000 0000_H*1	<RS-CAN0_base> + $093C_H$	8, 16, 32
Receive buffer ID register 52	RSCAN0RMID52	0000 0000_H*1	<RS-CAN0_base> + 0940_H	8, 16, 32
Receive buffer pointer register 52	RSCAN0RMPTR52	0000 0000_H*1	<RS-CAN0_base> + 0944_H	8, 16, 32

Table 16.10 List of RS-CAN Module Registers (12/21)

Register Name	Symbol	Value after Reset	Address	Access Size
Receive buffer data field 0 register 52	RSCAN0RMDF052	0000 0000_H*1	<RS-CAN0_base> + 0948_H	8, 16, 32
Receive buffer data field 1 register 52	RSCAN0RMDF152	0000 0000_H*1	<RS-CAN0_base> + $094C_H$	8, 16, 32
Receive buffer ID register 53	RSCAN0RMID53	0000 0000_H*1	<RS-CAN0_base> + 0950_H	8, 16, 32
Receive buffer pointer register 53	RSCAN0RMPTR53	0000 0000_H*1	<RS-CAN0_base> + 0954_H	8, 16, 32
Receive buffer data field 0 register 53	RSCAN0RMDF053	0000 0000_H*1	<RS-CAN0_base> + 0958_H	8, 16, 32
Receive buffer data field 1 register 53	RSCAN0RMDF153	0000 0000_H*1	<RS-CAN0_base> + $095C_H$	8, 16, 32
Receive buffer ID register 54	RSCAN0RMID54	0000 0000_H*1	<RS-CAN0_base> + 0960_H	8, 16, 32
Receive buffer pointer register 54	RSCAN0RMPTR54	0000 0000_H*1	<RS-CAN0_base> + 0964_H	8, 16, 32
Receive buffer data field 0 register 54	RSCAN0RMDF054	0000 0000_H*1	<RS-CAN0_base> + 0968_H	8, 16, 32
Receive buffer data field 1 register 54	RSCAN0RMDF154	0000 0000_H*1	<RS-CAN0_base> + $096C_H$	8, 16, 32
Receive buffer ID register 55	RSCAN0RMID55	0000 0000_H*1	<RS-CAN0_base> + 0970_H	8, 16, 32
Receive buffer pointer register 55	RSCAN0RMPTR55	0000 0000_H*1	<RS-CAN0_base> + 0974_H	8, 16, 32
Receive buffer data field 0 register 55	RSCAN0RMDF055	0000 0000_H*1	<RS-CAN0_base> + 0978_H	8, 16, 32
Receive buffer data field 1 register 55	RSCAN0RMDF155	0000 0000_H*1	<RS-CAN0_base> + $097C_H$	8, 16, 32
Receive buffer ID register 56	RSCAN0RMID56	0000 0000_H*1	<RS-CAN0_base> + 0980_H	8, 16, 32
Receive buffer pointer register 56	RSCAN0RMPTR56	0000 0000_H*1	<RS-CAN0_base> + 0984_H	8, 16, 32
Receive buffer data field 0 register 56	RSCAN0RMDF056	0000 0000_H*1	<RS-CAN0_base> + 0988_H	8, 16, 32
Receive buffer data field 1 register 56	RSCAN0RMDF156	0000 0000_H*1	<RS-CAN0_base> + $098C_H$	8, 16, 32
Receive buffer ID register 57	RSCAN0RMID57	0000 0000_H*1	<RS-CAN0_base> + 0990_H	8, 16, 32
Receive buffer pointer register 57	RSCAN0RMPTR57	0000 0000_H*1	<RS-CAN0_base> + 0994_H	8, 16, 32
Receive buffer data field 0 register 57	RSCAN0RMDF057	0000 0000_H*1	<RS-CAN0_base> + 0998_H	8, 16, 32
Receive buffer data field 1 register 57	RSCAN0RMDF157	0000 0000_H*1	<RS-CAN0_base> + $099C_H$	8, 16, 32
Receive buffer ID register 58	RSCAN0RMID58	0000 0000_H*1	<RS-CAN0_base> + $09A0_H$	8, 16, 32
Receive buffer pointer register 58	RSCAN0RMPTR58	0000 0000_H*1	<RS-CAN0_base> + $09A4_H$	8, 16, 32
Receive buffer data field 0 register 58	RSCAN0RMDF058	0000 0000_H*1	<RS-CAN0_base> + $09A8_H$	8, 16, 32
Receive buffer data field 1 register 58	RSCAN0RMDF158	0000 0000_H*1	<RS-CAN0_base> + $09AC_H$	8, 16, 32
Receive buffer ID register 59	RSCAN0RMID59	0000 0000_H*1	<RS-CAN0_base> + $09B0_H$	8, 16, 32
Receive buffer pointer register 59	RSCAN0RMPTR59	0000 0000_H*1	<RS-CAN0_base> + $09B4_H$	8, 16, 32
Receive buffer data field 0 register 59	RSCAN0RMDF059	0000 0000_H*1	<RS-CAN0_base> + $09B8_H$	8, 16, 32
Receive buffer data field 1 register 59	RSCAN0RMDF159	0000 0000_H*1	<RS-CAN0_base> + $09BC_H$	8, 16, 32
Receive buffer ID register 60	RSCAN0RMID60	0000 0000_H*1	<RS-CAN0_base> + $09C0_H$	8, 16, 32
Receive buffer pointer register 60	RSCAN0RMPTR60	0000 0000_H*1	<RS-CAN0_base> + $09C4_H$	8, 16, 32
Receive buffer data field 0 register 60	RSCAN0RMDF060	0000 0000_H*1	<RS-CAN0_base> + $09C8_H$	8, 16, 32
Receive buffer data field 1 register 60	RSCAN0RMDF160	0000 0000_H*1	<RS-CAN0_base> + $09CC_H$	8, 16, 32
Receive buffer ID register 61	RSCAN0RMID61	0000 0000_H*1	<RS-CAN0_base> + $09D0_H$	8, 16, 32
Receive buffer pointer register 61	RSCAN0RMPTR61	0000 0000_H*1	<RS-CAN0_base> + $09D4_H$	8, 16, 32
Receive buffer data field 0 register 61	RSCAN0RMDF061	0000 0000_H*1	<RS-CAN0_base> + $09D8_H$	8, 16, 32
Receive buffer data field 1 register 61	RSCAN0RMDF161	0000 0000_H*1	<RS-CAN0_base> + $09DC_H$	8, 16, 32
Receive buffer ID register 62	RSCAN0RMID62	0000 0000_H*1	<RS-CAN0_base> + $09E0_H$	8, 16, 32
Receive buffer pointer register 62	RSCAN0RMPTR62	0000 0000_H*1	<RS-CAN0_base> + $09E4_H$	8, 16, 32
Receive buffer data field 0 register 62	RSCAN0RMDF062	0000 0000_H*1	<RS-CAN0_base> + $09E8_H$	8, 16, 32
Receive buffer data field 1 register 62	RSCAN0RMDF162	0000 0000_H*1	<RS-CAN0_base> + $09EC_H$	8, 16, 32
Receive buffer ID register 63	RSCAN0RMID63	0000 0000_H*1	<RS-CAN0_base> + $09F0_H$	8, 16, 32
Receive buffer pointer register 63	RSCAN0RMPTR63	0000 0000_H*1	<RS-CAN0_base> + $09F4_H$	8, 16, 32
Receive buffer data field 0 register 63	RSCAN0RMDF063	0000 0000_H*1	<RS-CAN0_base> + $09F8_H$	8, 16, 32
Receive buffer data field 1 register 63	RSCAN0RMDF163	0000 0000_H*1	<RS-CAN0_base> + $09FC_H$	8, 16, 32
Receive FIFO access ID register 0	RSCAN0RFID0	0000 0000_H*1	<RS-CAN0_base> + $0E00_H$	8, 16, 32
Receive FIFO access pointer register 0	RSCAN0RFPTR0	0000 0000_H*1	<RS-CAN0_base> + $0E04_H$	8, 16, 32
Receive FIFO access data field 0 register 0	RSCAN0RFDF00	0000 0000_H*1	<RS-CAN0_base> + $0E08_H$	8, 16, 32

Table 16.10 List of RS-CAN Module Registers (13/21)

Register Name	Symbol	Value after Reset	Address	Access Size
Receive FIFO access data field 1 register 0	RSCAN0RFDF10	0000 0000$_H$*1	<RS-CAN0_base> + 0E0C$_H$	8, 16, 32
Receive FIFO access ID register 1	RSCAN0RFID1	0000 0000$_H$*1	<RS-CAN0_base> + 0E10$_H$	8, 16, 32
Receive FIFO access pointer register 1	RSCAN0RFPTR1	0000 0000$_H$*1	<RS-CAN0_base> + 0E14$_H$	8, 16, 32
Receive FIFO access data field 0 register 1	RSCAN0RFDF01	0000 0000$_H$*1	<RS-CAN0_base> + 0E18$_H$	8, 16, 32
Receive FIFO access data field 1 register 1	RSCAN0RFDF11	0000 0000$_H$*1	<RS-CAN0_base> + 0E1C$_H$	8, 16, 32
Receive FIFO access ID register 2	RSCAN0RFID2	0000 0000$_H$*1	<RS-CAN0_base> + 0E20$_H$	8, 16, 32
Receive FIFO access pointer register 2	RSCAN0RFPTR2	0000 0000$_H$*1	<RS-CAN0_base> + 0E24$_H$	8, 16, 32
Receive FIFO access data field 0 register 2	RSCAN0RFDF02	0000 0000$_H$*1	<RS-CAN0_base> + 0E28$_H$	8, 16, 32
Receive FIFO access data field 1 register 2	RSCAN0RFDF12	0000 0000$_H$*1	<RS-CAN0_base> + 0E2C$_H$	8, 16, 32
Receive FIFO access ID register 3	RSCAN0RFID3	0000 0000$_H$*1	<RS-CAN0_base> + 0E30$_H$	8, 16, 32
Receive FIFO access pointer register 3	RSCAN0RFPTR3	0000 0000$_H$*1	<RS-CAN0_base> + 0E34$_H$	8, 16, 32
Receive FIFO access data field 0 register 3	RSCAN0RFDF03	0000 0000$_H$*1	<RS-CAN0_base> + 0E38$_H$	8, 16, 32
Receive FIFO access data field 1 register 3	RSCAN0RFDF13	0000 0000$_H$*1	<RS-CAN0_base> + 0E3C$_H$	8, 16, 32
Receive FIFO access ID register 4	RSCAN0RFID4	0000 0000$_H$*1	<RS-CAN0_base> + 0E40$_H$	8, 16, 32
Receive FIFO access pointer register 4	RSCAN0RFPTR4	0000 0000$_H$*1	<RS-CAN0_base> + 0E44$_H$	8, 16, 32
Receive FIFO access data field 0 register 4	RSCAN0RFDF04	0000 0000$_H$*1	<RS-CAN0_base> + 0E48$_H$	8, 16, 32
Receive FIFO access data field 1 register 4	RSCAN0RFDF14	0000 0000$_H$*1	<RS-CAN0_base> + 0E4C$_H$	8, 16, 32
Receive FIFO access ID register 5	RSCAN0RFID5	0000 0000$_H$*1	<RS-CAN0_base> + 0E50$_H$	8, 16, 32
Receive FIFO access pointer register 5	RSCAN0RFPTR5	0000 0000$_H$*1	<RS-CAN0_base> + 0E54$_H$	8, 16, 32
Receive FIFO access data field 0 register 5	RSCAN0RFDF05	0000 0000$_H$*1	<RS-CAN0_base> + 0E58$_H$	8, 16, 32
Receive FIFO access data field 1 register 5	RSCAN0RFDF15	0000 0000$_H$*1	<RS-CAN0_base> + 0E5C$_H$	8, 16, 32
Receive FIFO access ID register 6	RSCAN0RFID6	0000 0000$_H$*1	<RS-CAN0_base> + 0E60$_H$	8, 16, 32
Receive FIFO access pointer register 6	RSCAN0RFPTR6	0000 0000$_H$*1	<RS-CAN0_base> + 0E64$_H$	8, 16, 32
Receive FIFO access data field 0 register 6	RSCAN0RFDF06	0000 0000$_H$*1	<RS-CAN0_base> + 0E68$_H$	8, 16, 32
Receive FIFO access data field 1 register 6	RSCAN0RFDF16	0000 0000$_H$*1	<RS-CAN0_base> + 0E6C$_H$	8, 16, 32
Receive FIFO access ID register 7	RSCAN0RFID7	0000 0000$_H$*1	<RS-CAN0_base> + 0E70$_H$	8, 16, 32
Receive FIFO access pointer register 7	RSCAN0RFPTR7	0000 0000$_H$*1	<RS-CAN0_base> + 0E74$_H$	8, 16, 32
Receive FIFO access data field 0 register 7	RSCAN0RFDF07	0000 0000$_H$*1	<RS-CAN0_base> + 0E78$_H$	8, 16, 32
Receive FIFO access data field 1 register 7	RSCAN0RFDF17	0000 0000$_H$*1	<RS-CAN0_base> + 0E7C$_H$	8, 16, 32
Transmit/receive FIFO access ID register 0	RSCAN0CFID0	0000 0000$_H$*1	<RS-CAN0_base> + 0E80$_H$	8, 16, 32
Transmit/receive FIFO access pointer register 0	RSCAN0CFPTR0	0000 0000$_H$*1	<RS-CAN0_base> + 0E84$_H$	8, 16, 32
Transmit/receive FIFO access data field 0 register 0	RSCAN0CFDF00	0000 0000$_H$*1	<RS-CAN0_base> + 0E88$_H$	8, 16, 32
Transmit/receive FIFO data field 1 register 0	RSCAN0CFDF10	0000 0000$_H$*1	<RS-CAN0_base> + 0E8C$_H$	8, 16, 32
Transmit/receive FIFO access ID register 1	RSCAN0CFID1	0000 0000$_H$*1	<RS-CAN0_base> + 0E90$_H$	8, 16, 32
Transmit/receive FIFO access pointer register 1	RSCAN0CFPTR1	0000 0000$_H$*1	<RS-CAN0_base> + 0E94$_H$	8, 16, 32
Transmit/receive FIFO access data field 0 register 1	RSCAN0CFDF01	0000 0000$_H$*1	<RS-CAN0_base> + 0E98$_H$	8, 16, 32
Transmit/receive FIFO data field 1 register 1	RSCAN0CFDF11	0000 0000$_H$*1	<RS-CAN0_base> + 0E9C$_H$	8, 16, 32
Transmit/receive FIFO access ID register 2	RSCAN0CFID2	0000 0000$_H$*1	<RS-CAN0_base> + 0EA0$_H$	8, 16, 32
Transmit/receive FIFO access pointer register 2	RSCAN0CFPTR2	0000 0000$_H$*1	<RS-CAN0_base> + 0EA4$_H$	8, 16, 32
Transmit/receive FIFO access data field 0 register 2	RSCAN0CFDF02	0000 0000$_H$*1	<RS-CAN0_base> + 0EA8$_H$	8, 16, 32
Transmit/receive FIFO data field 1 register 2	RSCAN0CFDF12	0000 0000$_H$*1	<RS-CAN0_base> + 0EAC$_H$	8, 16, 32
Transmit/receive FIFO access ID register 3	RSCAN0CFID3	0000 0000$_H$*1	<RS-CAN0_base> + 0EB0$_H$	8, 16, 32
Transmit/receive FIFO access pointer register 3	RSCAN0CFPTR3	0000 0000$_H$*1	<RS-CAN0_base> + 0EB4$_H$	8, 16, 32
Transmit/receive FIFO access data field 0 register 3	RSCAN0CFDF03	0000 0000$_H$*1	<RS-CAN0_base> + 0EB8$_H$	8, 16, 32
Transmit/receive FIFO data field 1 register 3	RSCAN0CFDF13	0000 0000$_H$*1	<RS-CAN0_base> + 0EBC$_H$	8, 16, 32
Transmit/receive FIFO access ID register 4	RSCAN0CFID4	0000 0000$_H$*1	<RS-CAN0_base> + 0EC0$_H$	8, 16, 32
Transmit/receive FIFO access pointer register 4	RSCAN0CFPTR4	0000 0000$_H$*1	<RS-CAN0_base> + 0EC4$_H$	8, 16, 32
Transmit/receive FIFO access data field 0 register 4	RSCAN0CFDF04	0000 0000$_H$*1	<RS-CAN0_base> + 0EC8$_H$	8, 16, 32
Transmit/receive FIFO data field 1 register 4	RSCAN0CFDF14	0000 0000$_H$*1	<RS-CAN0_base> + 0ECC$_H$	8, 16, 32

Table 16.10 List of RS-CAN Module Registers (14/21)

Register Name	Symbol	Value after Reset	Address	Access Size
Transmit/receive FIFO access ID register 5	RSCAN0CFID5	0000 0000$_H$*1	<RS-CAN0_base> + 0ED0$_H$	8, 16, 32
Transmit/receive FIFO access pointer register 5	RSCAN0CFPTR5	0000 0000$_H$*1	<RS-CAN0_base> + 0ED4$_H$	8, 16, 32
Transmit/receive FIFO access data field 0 register 5	RSCAN0CFDF05	0000 0000$_H$*1	<RS-CAN0_base> + 0ED8$_H$	8, 16, 32
Transmit/receive FIFO data field 1 register 5	RSCAN0CFDF15	0000 0000$_H$*1	<RS-CAN0_base> + 0EDC$_H$	8, 16, 32
Transmit/receive FIFO access ID register 6	RSCAN0CFID6	0000 0000$_H$*1	<RS-CAN0_base> + 0EE0$_H$	8, 16, 32
Transmit/receive FIFO access pointer register 6	RSCAN0CFPTR6	0000 0000$_H$*1	<RS-CAN0_base> + 0EE4$_H$	8, 16, 32
Transmit/receive FIFO access data field 0 register 6	RSCAN0CFDF06	0000 0000$_H$*1	<RS-CAN0_base> + 0EE8$_H$	8, 16, 32
Transmit/receive FIFO data field 1 register 6	RSCAN0CFDF16	0000 0000$_H$*1	<RS-CAN0_base> + 0EEC$_H$	8, 16, 32
Transmit/receive FIFO access ID register 7	RSCAN0CFID7	0000 0000$_H$*1	<RS-CAN0_base> + 0EF0$_H$	8, 16, 32
Transmit/receive FIFO access pointer register 7	RSCAN0CFPTR7	0000 0000$_H$*1	<RS-CAN0_base> + 0EF4$_H$	8, 16, 32
Transmit/receive FIFO access data field 0 register 7	RSCAN0CFDF07	0000 0000$_H$*1	<RS-CAN0_base> + 0EF8$_H$	8, 16, 32
Transmit/receive FIFO data field 1 register 7	RSCAN0CFDF17	0000 0000$_H$*1	<RS-CAN0_base> + 0EFC$_H$	8, 16, 32
Transmit/receive FIFO access ID register 8	RSCAN0CFID8	0000 0000$_H$*1	<RS-CAN0_base> + 0F00$_H$	8, 16, 32
Transmit/receive FIFO access pointer register 8	RSCAN0CFPTR8	0000 0000$_H$*1	<RS-CAN0_base> + 0F04$_H$	8, 16, 32
Transmit/receive FIFO access data field 0 register 8	RSCAN0CFDF08	0000 0000$_H$*1	<RS-CAN0_base> + 0F08$_H$	8, 16, 32
Transmit/receive FIFO data field 1 register 8	RSCAN0CFDF18	0000 0000$_H$*1	<RS-CAN0_base> + 0F0C$_H$	8, 16, 32
Transmit/receive FIFO access ID register 9	RSCAN0CFID9	0000 0000$_H$*1	<RS-CAN0_base> + 0F10$_H$	8, 16, 32
Transmit/receive FIFO access pointer register 9	RSCAN0CFPTR9	0000 0000$_H$*1	<RS-CAN0_base> + 0F14$_H$	8, 16, 32
Transmit/receive FIFO access data field 0 register 9	RSCAN0CFDF09	0000 0000$_H$*1	<RS-CAN0_base> + 0F18$_H$	8, 16, 32
Transmit/receive FIFO data field 1 register 9	RSCAN0CFDF19	0000 0000$_H$*1	<RS-CAN0_base> + 0F1C$_H$	8, 16, 32
Transmit/receive FIFO access ID register 10	RSCAN0CFID10	0000 0000$_H$*1	<RS-CAN0_base> + 0F20$_H$	8, 16, 32
Transmit/receive FIFO access pointer register 10	RSCAN0CFPTR10	0000 0000$_H$*1	<RS-CAN0_base> + 0F24$_H$	8, 16, 32
Transmit/receive FIFO access data field 0 register 10	RSCAN0CFDF010	0000 0000$_H$*1	<RS-CAN0_base> + 0F28$_H$	8, 16, 32
Transmit/receive FIFO data field 1 register 10	RSCAN0CFDF110	0000 0000$_H$*1	<RS-CAN0_base> + 0F2C$_H$	8, 16, 32
Transmit/receive FIFO access ID register 11	RSCAN0CFID11	0000 0000$_H$*1	<RS-CAN0_base> + 0F30$_H$	8, 16, 32
Transmit/receive FIFO access pointer register 11	RSCAN0CFPTR11	0000 0000$_H$*1	<RS-CAN0_base> + 0F34$_H$	8, 16, 32
Transmit/receive FIFO access data field 0 register 11	RSCAN0CFDF011	0000 0000$_H$*1	<RS-CAN0_base> + 0F38$_H$	8, 16, 32
Transmit/receive FIFO data field 1 register 11	RSCAN0CFDF111	0000 0000$_H$*1	<RS-CAN0_base> + 0F3C$_H$	8, 16, 32
Transmit buffer ID register 0	RSCAN0TMID0	0000 0000$_H$*1	<RS-CAN0_base> + 1000$_H$	8, 16, 32
Transmit buffer pointer register 0	RSCAN0TMPTR0	0000 0000$_H$*1	<RS-CAN0_base> + 1004$_H$	8, 16, 32
Transmit buffer data field 0 register 0	RSCAN0TMDF00	0000 0000$_H$*1	<RS-CAN0_base> + 1008$_H$	8, 16, 32
Transmit buffer data field 1 register 0	RSCAN0TMDF10	0000 0000$_H$*1	<RS-CAN0_base> + 100C$_H$	8, 16, 32
Transmit buffer ID register 1	RSCAN0TMID1	0000 0000$_H$*1	<RS-CAN0_base> + 1010$_H$	8, 16, 32
Transmit buffer pointer register 1	RSCAN0TMPTR1	0000 0000$_H$*1	<RS-CAN0_base> + 1014$_H$	8, 16, 32
Transmit buffer data field 0 register 1	RSCAN0TMDF01	0000 0000$_H$*1	<RS-CAN0_base> + 1018$_H$	8, 16, 32
Transmit buffer data field 1 register 1	RSCAN0TMDF11	0000 0000$_H$*1	<RS-CAN0_base> + 101C$_H$	8, 16, 32
Transmit buffer ID register 2	RSCAN0TMID2	0000 0000$_H$*1	<RS-CAN0_base> + 1020$_H$	8, 16, 32
Transmit buffer pointer register 2	RSCAN0TMPTR2	0000 0000$_H$*1	<RS-CAN0_base> + 1024$_H$	8, 16, 32
Transmit buffer data field 0 register 2	RSCAN0TMDF02	0000 0000$_H$*1	<RS-CAN0_base> + 1028$_H$	8, 16, 32
Transmit buffer data field 1 register 2	RSCAN0TMDF12	0000 0000$_H$*1	<RS-CAN0_base> + 102C$_H$	8, 16, 32
Transmit buffer ID register 3	RSCAN0TMID3	0000 0000$_H$*1	<RS-CAN0_base> + 1030$_H$	8, 16, 32
Transmit buffer pointer register 3	RSCAN0TMPTR3	0000 0000$_H$*1	<RS-CAN0_base> + 1034$_H$	8, 16, 32
Transmit buffer data field 0 register 3	RSCAN0TMDF03	0000 0000$_H$*1	<RS-CAN0_base> + 1038$_H$	8, 16, 32
Transmit buffer data field 1 register 3	RSCAN0TMDF13	0000 0000$_H$*1	<RS-CAN0_base> + 103C$_H$	8, 16, 32
Transmit buffer ID register 4	RSCAN0TMID4	0000 0000$_H$*1	<RS-CAN0_base> + 1040$_H$	8, 16, 32
Transmit buffer pointer register 4	RSCAN0TMPTR4	0000 0000$_H$*1	<RS-CAN0_base> + 1044$_H$	8, 16, 32
Transmit buffer data field 0 register 4	RSCAN0TMDF04	0000 0000$_H$*1	<RS-CAN0_base> + 1048$_H$	8, 16, 32
Transmit buffer data field 1 register 4	RSCAN0TMDF14	0000 0000$_H$*1	<RS-CAN0_base> + 104C$_H$	8, 16, 32
Transmit buffer ID register 5	RSCAN0TMID5	0000 0000$_H$*1	<RS-CAN0_base> + 1050$_H$	8, 16, 32

Table 16.10 List of RS-CAN Module Registers (15/21)

Register Name	Symbol	Value after Reset	Address	Access Size
Transmit buffer pointer register 5	RSCAN0TMPTR5	0000 0000_H*1	<RS-CAN0_base> + 1054_H	8, 16, 32
Transmit buffer data field 0 register 5	RSCAN0TMDF05	0000 0000_H*1	<RS-CAN0_base> + 1058_H	8, 16, 32
Transmit buffer data field 1 register 5	RSCAN0TMDF15	0000 0000_H*1	<RS-CAN0_base> + $105C_H$	8, 16, 32
Transmit buffer ID register 6	RSCAN0TMID6	0000 0000_H*1	<RS-CAN0_base> + 1060_H	8, 16, 32
Transmit buffer pointer register 6	RSCAN0TMPTR6	0000 0000_H*1	<RS-CAN0_base> + 1064_H	8, 16, 32
Transmit buffer data field 0 register 6	RSCAN0TMDF06	0000 0000_H*1	<RS-CAN0_base> + 1068_H	8, 16, 32
Transmit buffer data field 1 register 6	RSCAN0TMDF16	0000 0000_H*1	<RS-CAN0_base> + $106C_H$	8, 16, 32
Transmit buffer ID register 7	RSCAN0TMID7	0000 0000_H*1	<RS-CAN0_base> + 1070_H	8, 16, 32
Transmit buffer pointer register 7	RSCAN0TMPTR7	0000 0000_H*1	<RS-CAN0_base> + 1074_H	8, 16, 32
Transmit buffer data field 0 register 7	RSCAN0TMDF07	0000 0000_H*1	<RS-CAN0_base> + 1078_H	8, 16, 32
Transmit buffer data field 1 register 7	RSCAN0TMDF17	0000 0000_H*1	<RS-CAN0_base> + $107C_H$	8, 16, 32
Transmit buffer ID register 8	RSCAN0TMID8	0000 0000_H*1	<RS-CAN0_base> + 1080_H	8, 16, 32
Transmit buffer pointer register 8	RSCAN0TMPTR8	0000 0000_H*1	<RS-CAN0_base> + 1084_H	8, 16, 32
Transmit buffer data field 0 register 8	RSCAN0TMDF08	0000 0000_H*1	<RS-CAN0_base> + 1088_H	8, 16, 32
Transmit buffer data field 1 register 8	RSCAN0TMDF18	0000 0000_H*1	<RS-CAN0_base> + $108C_H$	8, 16, 32
Transmit buffer ID register 9	RSCAN0TMID9	0000 0000_H*1	<RS-CAN0_base> + 1090_H	8, 16, 32
Transmit buffer pointer register 9	RSCAN0TMPTR9	0000 0000_H*1	<RS-CAN0_base> + 1094_H	8, 16, 32
Transmit buffer data field 0 register 9	RSCAN0TMDF09	0000 0000_H*1	<RS-CAN0_base> + 1098_H	8, 16, 32
Transmit buffer data field 1 register 9	RSCAN0TMDF19	0000 0000_H*1	<RS-CAN0_base> + $109C_H$	8, 16, 32
Transmit buffer ID register 10	RSCAN0TMID10	0000 0000_H*1	<RS-CAN0_base> + $10A0_H$	8, 16, 32
Transmit buffer pointer register 10	RSCAN0TMPTR10	0000 0000_H*1	<RS-CAN0_base> + $10A4_H$	8, 16, 32
Transmit buffer data field 0 register 10	RSCAN0TMDF010	0000 0000_H*1	<RS-CAN0_base> + $10A8_H$	8, 16, 32
Transmit buffer data field 1 register 10	RSCAN0TMDF110	0000 0000_H*1	<RS-CAN0_base> + $10AC_H$	8, 16, 32
Transmit buffer ID register 11	RSCAN0TMID11	0000 0000_H*1	<RS-CAN0_base> + $10B0_H$	8, 16, 32
Transmit buffer pointer register 11	RSCAN0TMPTR11	0000 0000_H*1	<RS-CAN0_base> + $10B4_H$	8, 16, 32
Transmit buffer data field 0 register 11	RSCAN0TMDF011	0000 0000_H*1	<RS-CAN0_base> + $10B8_H$	8, 16, 32
Transmit buffer data field 1 register 11	RSCAN0TMDF111	0000 0000_H*1	<RS-CAN0_base> + $10BC_H$	8, 16, 32
Transmit buffer ID register 12	RSCAN0TMID12	0000 0000_H*1	<RS-CAN0_base> + $10C0_H$	8, 16, 32
Transmit buffer pointer register 12	RSCAN0TMPTR12	0000 0000_H*1	<RS-CAN0_base> + $10C4_H$	8, 16, 32
Transmit buffer data field 0 register 12	RSCAN0TMDF012	0000 0000_H*1	<RS-CAN0_base> + $10C8_H$	8, 16, 32
Transmit buffer data field 1 register 12	RSCAN0TMDF112	0000 0000_H*1	<RS-CAN0_base> + $10CC_H$	8, 16, 32
Transmit buffer ID register 13	RSCAN0TMID13	0000 0000_H*1	<RS-CAN0_base> + $10D0_H$	8, 16, 32
Transmit buffer pointer register 13	RSCAN0TMPTR13	0000 0000_H*1	<RS-CAN0_base> + $10D4_H$	8, 16, 32
Transmit buffer data field 0 register 13	RSCAN0TMDF013	0000 0000_H*1	<RS-CAN0_base> + $10D8_H$	8, 16, 32
Transmit buffer data field 1 register 13	RSCAN0TMDF113	0000 0000_H*1	<RS-CAN0_base> + $10DC_H$	8, 16, 32
Transmit buffer ID register 14	RSCAN0TMID14	0000 0000_H*1	<RS-CAN0_base> + $10E0_H$	8, 16, 32
Transmit buffer pointer register 14	RSCAN0TMPTR14	0000 0000_H*1	<RS-CAN0_base> + $10E4_H$	8, 16, 32
Transmit buffer data field 0 register 14	RSCAN0TMDF014	0000 0000_H*1	<RS-CAN0_base> + $10E8_H$	8, 16, 32
Transmit buffer data field 1 register 14	RSCAN0TMDF114	0000 0000_H*1	<RS-CAN0_base> + $10EC_H$	8, 16, 32
Transmit buffer ID register 15	RSCAN0TMID15	0000 0000_H*1	<RS-CAN0_base> + $10F0_H$	8, 16, 32
Transmit buffer pointer register 15	RSCAN0TMPTR15	0000 0000_H*1	<RS-CAN0_base> + $10F4_H$	8, 16, 32
Transmit buffer data field 0 register 15	RSCAN0TMDF015	0000 0000_H*1	<RS-CAN0_base> + $10F8_H$	8, 16, 32
Transmit buffer data field 1 register 15	RSCAN0TMDF115	0000 0000_H*1	<RS-CAN0_base> + $10FC_H$	8, 16, 32
Transmit buffer ID register 16	RSCAN0TMID16	0000 0000_H*1	<RS-CAN0_base> + 1100_H	8, 16, 32
Transmit buffer pointer register 16	RSCAN0TMPTR16	0000 0000_H*1	<RS-CAN0_base> + 1104_H	8, 16, 32
Transmit buffer data field 0 register 16	RSCAN0TMDF016	0000 0000_H*1	<RS-CAN0_base> + 1108_H	8, 16, 32
Transmit buffer data field 1 register 16	RSCAN0TMDF116	0000 0000_H*1	<RS-CAN0_base> + $110C_H$	8, 16, 32
Transmit buffer ID register 17	RSCAN0TMID17	0000 0000_H*1	<RS-CAN0_base> + 1110_H	8, 16, 32
Transmit buffer pointer register 17	RSCAN0TMPTR17	0000 0000_H*1	<RS-CAN0_base> + 1114_H	8, 16, 32

Table 16.10 List of RS-CAN Module Registers (16/21)

Register Name	Symbol	Value after Reset	Address	Access Size
Transmit buffer data field 0 register 17	RSCAN0TMDF017	0000 0000_H*1	<RS-CAN0_base> + 1118_H	8, 16, 32
Transmit buffer data field 1 register 17	RSCAN0TMDF117	0000 0000_H*1	<RS-CAN0_base> + $111C_H$	8, 16, 32
Transmit buffer ID register 18	RSCAN0TMID18	0000 0000_H*1	<RS-CAN0_base> + 1120_H	8, 16, 32
Transmit buffer pointer register 18	RSCAN0TMPTR18	0000 0000_H*1	<RS-CAN0_base> + 1124_H	8, 16, 32
Transmit buffer data field 0 register 18	RSCAN0TMDF018	0000 0000_H*1	<RS-CAN0_base> + 1128_H	8, 16, 32
Transmit buffer data field 1 register 18	RSCAN0TMDF118	0000 0000_H*1	<RS-CAN0_base> + $112C_H$	8, 16, 32
Transmit buffer ID register 19	RSCAN0TMID19	0000 0000_H*1	<RS-CAN0_base> + 1130_H	8, 16, 32
Transmit buffer pointer register 19	RSCAN0TMPTR19	0000 0000_H*1	<RS-CAN0_base> + 1134_H	8, 16, 32
Transmit buffer data field 0 register 19	RSCAN0TMDF019	0000 0000_H*1	<RS-CAN0_base> + 1138_H	8, 16, 32
Transmit buffer data field 1 register 19	RSCAN0TMDF119	0000 0000_H*1	<RS-CAN0_base> + $113C_H$	8, 16, 32
Transmit buffer ID register 20	RSCAN0TMID20	0000 0000_H*1	<RS-CAN0_base> + 1140_H	8, 16, 32
Transmit buffer pointer register 20	RSCAN0TMPTR20	0000 0000_H*1	<RS-CAN0_base> + 1144_H	8, 16, 32
Transmit buffer data field 0 register 20	RSCAN0TMDF020	0000 0000_H*1	<RS-CAN0_base> + 1148_H	8, 16, 32
Transmit buffer data field 1 register 20	RSCAN0TMDF120	0000 0000_H*1	<RS-CAN0_base> + $114C_H$	8, 16, 32
Transmit buffer ID register 21	RSCAN0TMID21	0000 0000_H*1	<RS-CAN0_base> + 1150_H	8, 16, 32
Transmit buffer pointer register 21	RSCAN0TMPTR21	0000 0000_H*1	<RS-CAN0_base> + 1154_H	8, 16, 32
Transmit buffer data field 0 register 21	RSCAN0TMDF021	0000 0000_H*1	<RS-CAN0_base> + 1158_H	8, 16, 32
Transmit buffer data field 1 register 21	RSCAN0TMDF121	0000 0000_H*1	<RS-CAN0_base> + $115C_H$	8, 16, 32
Transmit buffer ID register 22	RSCAN0TMID22	0000 0000_H*1	<RS-CAN0_base> + 1160_H	8, 16, 32
Transmit buffer pointer register 22	RSCAN0TMPTR22	0000 0000_H*1	<RS-CAN0_base> + 1164_H	8, 16, 32
Transmit buffer data field 0 register 22	RSCAN0TMDF022	0000 0000_H*1	<RS-CAN0_base> + 1168_H	8, 16, 32
Transmit buffer data field 1 register 22	RSCAN0TMDF122	0000 0000_H*1	<RS-CAN0_base> + $116C_H$	8, 16, 32
Transmit buffer ID register 23	RSCAN0TMID23	0000 0000_H*1	<RS-CAN0_base> + 1170_H	8, 16, 32
Transmit buffer pointer register 23	RSCAN0TMPTR23	0000 0000_H*1	<RS-CAN0_base> + 1174_H	8, 16, 32
Transmit buffer data field 0 register 23	RSCAN0TMDF023	0000 0000_H*1	<RS-CAN0_base> + 1178_H	8, 16, 32
Transmit buffer data field 1 register 23	RSCAN0TMDF123	0000 0000_H*1	<RS-CAN0_base> + $117C_H$	8, 16, 32
Transmit buffer ID register 24	RSCAN0TMID24	0000 0000_H*1	<RS-CAN0_base> + 1180_H	8, 16, 32
Transmit buffer pointer register 24	RSCAN0TMPTR24	0000 0000_H*1	<RS-CAN0_base> + 1184_H	8, 16, 32
Transmit buffer data field 0 register 24	RSCAN0TMDF024	0000 0000_H*1	<RS-CAN0_base> + 1188_H	8, 16, 32
Transmit buffer data field 1 register 24	RSCAN0TMDF124	0000 0000_H*1	<RS-CAN0_base> + $118C_H$	8, 16, 32
Transmit buffer ID register 25	RSCAN0TMID25	0000 0000_H*1	<RS-CAN0_base> + 1190_H	8, 16, 32
Transmit buffer pointer register 25	RSCAN0TMPTR25	0000 0000_H*1	<RS-CAN0_base> + 1194_H	8, 16, 32
Transmit buffer data field 0 register 25	RSCAN0TMDF025	0000 0000_H*1	<RS-CAN0_base> + 1198_H	8, 16, 32
Transmit buffer data field 1 register 25	RSCAN0TMDF125	0000 0000_H*1	<RS-CAN0_base> + $119C_H$	8, 16, 32
Transmit buffer ID register 26	RSCAN0TMID26	0000 0000_H*1	<RS-CAN0_base> + $11A0_H$	8, 16, 32
Transmit buffer pointer register 26	RSCAN0TMPTR26	0000 0000_H*1	<RS-CAN0_base> + $11A4_H$	8, 16, 32
Transmit buffer data field 0 register 26	RSCAN0TMDF026	0000 0000_H*1	<RS-CAN0_base> + $11A8_H$	8, 16, 32
Transmit buffer data field 1 register 26	RSCAN0TMDF126	0000 0000_H*1	<RS-CAN0_base> + $11AC_H$	8, 16, 32
Transmit buffer ID register 27	RSCAN0TMID27	0000 0000_H*1	<RS-CAN0_base> + $11B0_H$	8, 16, 32
Transmit buffer pointer register 27	RSCAN0TMPTR27	0000 0000_H*1	<RS-CAN0_base> + $11B4_H$	8, 16, 32
Transmit buffer data field 0 register 27	RSCAN0TMDF027	0000 0000_H*1	<RS-CAN0_base> + $11B8_H$	8, 16, 32
Transmit buffer data field 1 register 27	RSCAN0TMDF127	0000 0000_H*1	<RS-CAN0_base> + $11BC_H$	8, 16, 32
Transmit buffer ID register 28	RSCAN0TMID28	0000 0000_H*1	<RS-CAN0_base> + $11C0_H$	8, 16, 32
Transmit buffer pointer register 28	RSCAN0TMPTR28	0000 0000_H*1	<RS-CAN0_base> + $11C4_H$	8, 16, 32
Transmit buffer data field 0 register 28	RSCAN0TMDF028	0000 0000_H*1	<RS-CAN0_base> + $11C8_H$	8, 16, 32
Transmit buffer data field 1 register 28	RSCAN0TMDF128	0000 0000_H*1	<RS-CAN0_base> + $11CC_H$	8, 16, 32
Transmit buffer ID register 29	RSCAN0TMID29	0000 0000_H*1	<RS-CAN0_base> + $11D0_H$	8, 16, 32
Transmit buffer pointer register 29	RSCAN0TMPTR29	0000 0000_H*1	<RS-CAN0_base> + $11D4_H$	8, 16, 32
Transmit buffer data field 0 register 29	RSCAN0TMDF029	0000 0000_H*1	<RS-CAN0_base> + $11D8_H$	8, 16, 32

Table 16.10 List of RS-CAN Module Registers (17/21)

Register Name	Symbol	Value after Reset	Address	Access Size
Transmit buffer data field 1 register 29	RSCAN0TMDF129	0000 0000_H*1	<RS-CAN0_base> + $11DC_H$	8, 16, 32
Transmit buffer ID register 30	RSCAN0TMID30	0000 0000_H*1	<RS-CAN0_base> + $11E0_H$	8, 16, 32
Transmit buffer pointer register 30	RSCAN0TMPTR30	0000 0000_H*1	<RS-CAN0_base> + $11E4_H$	8, 16, 32
Transmit buffer data field 0 register 30	RSCAN0TMDF030	0000 0000_H*1	<RS-CAN0_base> + $11E8_H$	8, 16, 32
Transmit buffer data field 1 register 30	RSCAN0TMDF130	0000 0000_H*1	<RS-CAN0_base> + $11EC_H$	8, 16, 32
Transmit buffer ID register 31	RSCAN0TMID31	0000 0000_H*1	<RS-CAN0_base> + $11F0_H$	8, 16, 32
Transmit buffer pointer register 31	RSCAN0TMPTR31	0000 0000_H*1	<RS-CAN0_base> + $11F4_H$	8, 16, 32
Transmit buffer data field 0 register 31	RSCAN0TMDF031	0000 0000_H*1	<RS-CAN0_base> + $11F8_H$	8, 16, 32
Transmit buffer data field 1 register 31	RSCAN0TMDF131	0000 0000_H*1	<RS-CAN0_base> + $11FC_H$	8, 16, 32
Transmit buffer ID register 32	RSCAN0TMID32	0000 0000_H*1	<RS-CAN0_base> + 1200_H	8, 16, 32
Transmit buffer pointer register 32	RSCAN0TMPTR32	0000 0000_H*1	<RS-CAN0_base> + 1204_H	8, 16, 32
Transmit buffer data field 0 register 32	RSCAN0TMDF032	0000 0000_H*1	<RS-CAN0_base> + 1208_H	8, 16, 32
Transmit buffer data field 1 register 32	RSCAN0TMDF132	0000 0000_H*1	<RS-CAN0_base> + $120C_H$	8, 16, 32
Transmit buffer ID register 33	RSCAN0TMID33	0000 0000_H*1	<RS-CAN0_base> + 1210_H	8, 16, 32
Transmit buffer pointer register 33	RSCAN0TMPTR33	0000 0000_H*1	<RS-CAN0_base> + 1214_H	8, 16, 32
Transmit buffer data field 0 register 33	RSCAN0TMDF033	0000 0000_H*1	<RS-CAN0_base> + 1218_H	8, 16, 32
Transmit buffer data field 1 register 33	RSCAN0TMDF133	0000 0000_H*1	<RS-CAN0_base> + $121C_H$	8, 16, 32
Transmit buffer ID register 34	RSCAN0TMID34	0000 0000_H*1	<RS-CAN0_base> + 1220_H	8, 16, 32
Transmit buffer pointer register 34	RSCAN0TMPTR34	0000 0000_H*1	<RS-CAN0_base> + 1224_H	8, 16, 32
Transmit buffer data field 0 register 34	RSCAN0TMDF034	0000 0000_H*1	<RS-CAN0_base> + 1228_H	8, 16, 32
Transmit buffer data field 1 register 34	RSCAN0TMDF134	0000 0000_H*1	<RS-CAN0_base> + $122C_H$	8, 16, 32
Transmit buffer ID register 35	RSCAN0TMID35	0000 0000_H*1	<RS-CAN0_base> + 1230_H	8, 16, 32
Transmit buffer pointer register 35	RSCAN0TMPTR35	0000 0000_H*1	<RS-CAN0_base> + 1234_H	8, 16, 32
Transmit buffer data field 0 register 35	RSCAN0TMDF035	0000 0000_H*1	<RS-CAN0_base> + 1238_H	8, 16, 32
Transmit buffer data field 1 register 35	RSCAN0TMDF135	0000 0000_H*1	<RS-CAN0_base> + $123C_H$	8, 16, 32
Transmit buffer ID register 36	RSCAN0TMID36	0000 0000_H*1	<RS-CAN0_base> + 1240_H	8, 16, 32
Transmit buffer pointer register 36	RSCAN0TMPTR36	0000 0000_H*1	<RS-CAN0_base> + 1244_H	8, 16, 32
Transmit buffer data field 0 register 36	RSCAN0TMDF036	0000 0000_H*1	<RS-CAN0_base> + 1248_H	8, 16, 32
Transmit buffer data field 1 register 36	RSCAN0TMDF136	0000 0000_H*1	<RS-CAN0_base> + $124C_H$	8, 16, 32
Transmit buffer ID register 37	RSCAN0TMID37	0000 0000_H*1	<RS-CAN0_base> + 1250_H	8, 16, 32
Transmit buffer pointer register 37	RSCAN0TMPTR37	0000 0000_H*1	<RS-CAN0_base> + 1254_H	8, 16, 32
Transmit buffer data field 0 register 37	RSCAN0TMDF037	0000 0000_H*1	<RS-CAN0_base> + 1258_H	8, 16, 32
Transmit buffer data field 1 register 37	RSCAN0TMDF137	0000 0000_H*1	<RS-CAN0_base> + $125C_H$	8, 16, 32
Transmit buffer ID register 38	RSCAN0TMID38	0000 0000_H*1	<RS-CAN0_base> + 1260_H	8, 16, 32
Transmit buffer pointer register 38	RSCAN0TMPTR38	0000 0000_H*1	<RS-CAN0_base> + 1264_H	8, 16, 32
Transmit buffer data field 0 register 38	RSCAN0TMDF038	0000 0000_H*1	<RS-CAN0_base> + 1268_H	8, 16, 32
Transmit buffer data field 1 register 38	RSCAN0TMDF138	0000 0000_H*1	<RS-CAN0_base> + $126C_H$	8, 16, 32
Transmit buffer ID register 39	RSCAN0TMID39	0000 0000_H*1	<RS-CAN0_base> + 1270_H	8, 16, 32
Transmit buffer pointer register 39	RSCAN0TMPTR39	0000 0000_H*1	<RS-CAN0_base> + 1274_H	8, 16, 32
Transmit buffer data field 0 register 39	RSCAN0TMDF039	0000 0000_H*1	<RS-CAN0_base> + 1278_H	8, 16, 32
Transmit buffer data field 1 register 39	RSCAN0TMDF139	0000 0000_H*1	<RS-CAN0_base> + $127C_H$	8, 16, 32
Transmit buffer ID register 40	RSCAN0TMID40	0000 0000_H*1	<RS-CAN0_base> + 1280_H	8, 16, 32
Transmit buffer pointer register 40	RSCAN0TMPTR40	0000 0000_H*1	<RS-CAN0_base> + 1284_H	8, 16, 32
Transmit buffer data field 0 register 40	RSCAN0TMDF040	0000 0000_H*1	<RS-CAN0_base> + 1288_H	8, 16, 32
Transmit buffer data field 1 register 40	RSCAN0TMDF140	0000 0000_H*1	<RS-CAN0_base> + $128C_H$	8, 16, 32
Transmit buffer ID register 41	RSCAN0TMID41	0000 0000_H*1	<RS-CAN0_base> + 1290_H	8, 16, 32
Transmit buffer pointer register 41	RSCAN0TMPTR41	0000 0000_H*1	<RS-CAN0_base> + 1294_H	8, 16, 32
Transmit buffer data field 0 register 41	RSCAN0TMDF041	0000 0000_H*1	<RS-CAN0_base> + 1298_H	8, 16, 32
Transmit buffer data field 1 register 41	RSCAN0TMDF141	0000 0000_H*1	<RS-CAN0_base> + $129C_H$	8, 16, 32

Table 16.10 List of RS-CAN Module Registers (18/21)

Register Name	Symbol	Value after Reset	Address	Access Size
Transmit buffer ID register 42	RSCAN0TMID42	0000 0000_H*1	<RS-CAN0_base> + $12A0_H$	8, 16, 32
Transmit buffer pointer register 42	RSCAN0TMPTR42	0000 0000_H*1	<RS-CAN0_base> + $12A4_H$	8, 16, 32
Transmit buffer data field 0 register 42	RSCAN0TMDF042	0000 0000_H*1	<RS-CAN0_base> + $12A8_H$	8, 16, 32
Transmit buffer data field 1 register 42	RSCAN0TMDF142	0000 0000_H*1	<RS-CAN0_base> + $12AC_H$	8, 16, 32
Transmit buffer ID register 43	RSCAN0TMID43	0000 0000_H*1	<RS-CAN0_base> + $12B0_H$	8, 16, 32
Transmit buffer pointer register 43	RSCAN0TMPTR43	0000 0000_H*1	<RS-CAN0_base> + $12B4_H$	8, 16, 32
Transmit buffer data field 0 register 43	RSCAN0TMDF043	0000 0000_H*1	<RS-CAN0_base> + $12B8_H$	8, 16, 32
Transmit buffer data field 1 register 43	RSCAN0TMDF143	0000 0000_H*1	<RS-CAN0_base> + $12BC_H$	8, 16, 32
Transmit buffer ID register 44	RSCAN0TMID44	0000 0000_H*1	<RS-CAN0_base> + $12C0_H$	8, 16, 32
Transmit buffer pointer register 44	RSCAN0TMPTR44	0000 0000_H*1	<RS-CAN0_base> + $12C4_H$	8, 16, 32
Transmit buffer data field 0 register 44	RSCAN0TMDF044	0000 0000_H*1	<RS-CAN0_base> + $12C8_H$	8, 16, 32
Transmit buffer data field 1 register 44	RSCAN0TMDF144	0000 0000_H*1	<RS-CAN0_base> + $12CC_H$	8, 16, 32
Transmit buffer ID register 45	RSCAN0TMID45	0000 0000_H*1	<RS-CAN0_base> + $12D0_H$	8, 16, 32
Transmit buffer pointer register 45	RSCAN0TMPTR45	0000 0000_H*1	<RS-CAN0_base> + $12D4_H$	8, 16, 32
Transmit buffer data field 0 register 45	RSCAN0TMDF045	0000 0000_H*1	<RS-CAN0_base> + $12D8_H$	8, 16, 32
Transmit buffer data field 1 register 45	RSCAN0TMDF145	0000 0000_H*1	<RS-CAN0_base> + $12DC_H$	8, 16, 32
Transmit buffer ID register 46	RSCAN0TMID46	0000 0000_H*1	<RS-CAN0_base> + $12E0_H$	8, 16, 32
Transmit buffer pointer register 46	RSCAN0TMPTR46	0000 0000_H*1	<RS-CAN0_base> + $12E4_H$	8, 16, 32
Transmit buffer data field 0 register 46	RSCAN0TMDF046	0000 0000_H*1	<RS-CAN0_base> + $12E8_H$	8, 16, 32
Transmit buffer data field 1 register 46	RSCAN0TMDF146	0000 0000_H*1	<RS-CAN0_base> + $12EC_H$	8, 16, 32
Transmit buffer ID register 47	RSCAN0TMID47	0000 0000_H*1	<RS-CAN0_base> + $12F0_H$	8, 16, 32
Transmit buffer pointer register 47	RSCAN0TMPTR47	0000 0000_H*1	<RS-CAN0_base> + $12F4_H$	8, 16, 32
Transmit buffer data field 0 register 47	RSCAN0TMDF047	0000 0000_H*1	<RS-CAN0_base> + $12F8_H$	8, 16, 32
Transmit buffer data field 1 register 47	RSCAN0TMDF147	0000 0000_H*1	<RS-CAN0_base> + $12FC_H$	8, 16, 32
Transmit buffer ID register 48	RSCAN0TMID48	0000 0000_H*1	<RS-CAN0_base> + 1300_H	8, 16, 32
Transmit buffer pointer register 48	RSCAN0TMPTR48	0000 0000_H*1	<RS-CAN0_base> + 1304_H	8, 16, 32
Transmit buffer data field 0 register 48	RSCAN0TMDF048	0000 0000_H*1	<RS-CAN0_base> + 1308_H	8, 16, 32
Transmit buffer data field 1 register 48	RSCAN0TMDF148	0000 0000_H*1	<RS-CAN0_base> + $130C_H$	8, 16, 32
Transmit buffer ID register 49	RSCAN0TMID49	0000 0000_H*1	<RS-CAN0_base> + 1310_H	8, 16, 32
Transmit buffer pointer register 49	RSCAN0TMPTR49	0000 0000_H*1	<RS-CAN0_base> + 1314_H	8, 16, 32
Transmit buffer data field 0 register 49	RSCAN0TMDF049	0000 0000_H*1	<RS-CAN0_base> + 1318_H	8, 16, 32
Transmit buffer data field 1 register 49	RSCAN0TMDF149	0000 0000_H*1	<RS-CAN0_base> + $131C_H$	8, 16, 32
Transmit buffer ID register 50	RSCAN0TMID50	0000 0000_H*1	<RS-CAN0_base> + 1320_H	8, 16, 32
Transmit buffer pointer register 50	RSCAN0TMPTR50	0000 0000_H*1	<RS-CAN0_base> + 1324_H	8, 16, 32
Transmit buffer data field 0 register 50	RSCAN0TMDF050	0000 0000_H*1	<RS-CAN0_base> + 1328_H	8, 16, 32
Transmit buffer data field 1 register 50	RSCAN0TMDF150	0000 0000_H*1	<RS-CAN0_base> + $132C_H$	8, 16, 32
Transmit buffer ID register 51	RSCAN0TMID51	0000 0000_H*1	<RS-CAN0_base> + 1330_H	8, 16, 32
Transmit buffer pointer register 51	RSCAN0TMPTR51	0000 0000_H*1	<RS-CAN0_base> + 1334_H	8, 16, 32
Transmit buffer data field 0 register 51	RSCAN0TMDF051	0000 0000_H*1	<RS-CAN0_base> + 1338_H	8, 16, 32
Transmit buffer data field 1 register 51	RSCAN0TMDF151	0000 0000_H*1	<RS-CAN0_base> + $133C_H$	8, 16, 32
Transmit buffer ID register 52	RSCAN0TMID52	0000 0000_H*1	<RS-CAN0_base> + 1340_H	8, 16, 32
Transmit buffer pointer register 52	RSCAN0TMPTR52	0000 0000_H*1	<RS-CAN0_base> + 1344_H	8, 16, 32
Transmit buffer data field 0 register 52	RSCAN0TMDF052	0000 0000_H*1	<RS-CAN0_base> + 1348_H	8, 16, 32
Transmit buffer data field 1 register 52	RSCAN0TMDF152	0000 0000_H*1	<RS-CAN0_base> + $134C_H$	8, 16, 32
Transmit buffer ID register 53	RSCAN0TMID53	0000 0000_H*1	<RS-CAN0_base> + 1350_H	8, 16, 32
Transmit buffer pointer register 53	RSCAN0TMPTR53	0000 0000_H*1	<RS-CAN0_base> + 1354_H	8, 16, 32
Transmit buffer data field 0 register 53	RSCAN0TMDF053	0000 0000_H*1	<RS-CAN0_base> + 1358_H	8, 16, 32
Transmit buffer data field 1 register 53	RSCAN0TMDF153	0000 0000_H*1	<RS-CAN0_base> + $135C_H$	8, 16, 32
Transmit buffer ID register 54	RSCAN0TMID54	0000 0000_H*1	<RS-CAN0_base> + 1360_H	8, 16, 32

Table 16.10 List of RS-CAN Module Registers (19/21)

Register Name	Symbol	Value after Reset	Address	Access Size
Transmit buffer pointer register 54	RSCAN0TMPTR54	0000 0000_H*1	<RS-CAN0_base> + 1364_H	8, 16, 32
Transmit buffer data field 0 register 54	RSCAN0TMDF054	0000 0000_H*1	<RS-CAN0_base> + 1368_H	8, 16, 32
Transmit buffer data field 1 register 54	RSCAN0TMDF154	0000 0000_H*1	<RS-CAN0_base> + $136C_H$	8, 16, 32
Transmit buffer ID register 55	RSCAN0TMID55	0000 0000_H*1	<RS-CAN0_base> + 1370_H	8, 16, 32
Transmit buffer pointer register 55	RSCAN0TMPTR55	0000 0000_H*1	<RS-CAN0_base> + 1374_H	8, 16, 32
Transmit buffer data field 0 register 55	RSCAN0TMDF055	0000 0000_H*1	<RS-CAN0_base> + 1378_H	8, 16, 32
Transmit buffer data field 1 register 55	RSCAN0TMDF155	0000 0000_H*1	<RS-CAN0_base> + $137C_H$	8, 16, 32
Transmit buffer ID register 56	RSCAN0TMID56	0000 0000_H*1	<RS-CAN0_base> + 1380_H	8, 16, 32
Transmit buffer pointer register 56	RSCAN0TMPTR56	0000 0000_H*1	<RS-CAN0_base> + 1384_H	8, 16, 32
Transmit buffer data field 0 register 56	RSCAN0TMDF056	0000 0000_H*1	<RS-CAN0_base> + 1388_H	8, 16, 32
Transmit buffer data field 1 register 56	RSCAN0TMDF156	0000 0000_H*1	<RS-CAN0_base> + $138C_H$	8, 16, 32
Transmit buffer ID register 57	RSCAN0TMID57	0000 0000_H*1	<RS-CAN0_base> + 1390_H	8, 16, 32
Transmit buffer pointer register 57	RSCAN0TMPTR57	0000 0000_H*1	<RS-CAN0_base> + 1394_H	8, 16, 32
Transmit buffer data field 0 register 57	RSCAN0TMDF057	0000 0000_H*1	<RS-CAN0_base> + 1398_H	8, 16, 32
Transmit buffer data field 1 register 57	RSCAN0TMDF157	0000 0000_H*1	<RS-CAN0_base> + $139C_H$	8, 16, 32
Transmit buffer ID register 58	RSCAN0TMID58	0000 0000_H*1	<RS-CAN0_base> + $13A0_H$	8, 16, 32
Transmit buffer pointer register 58	RSCAN0TMPTR58	0000 0000_H*1	<RS-CAN0_base> + $13A4_H$	8, 16, 32
Transmit buffer data field 0 register 58	RSCAN0TMDF058	0000 0000_H*1	<RS-CAN0_base> + $13A8_H$	8, 16, 32
Transmit buffer data field 1 register 58	RSCAN0TMDF158	0000 0000_H*1	<RS-CAN0_base> + $13AC_H$	8, 16, 32
Transmit buffer ID register 59	RSCAN0TMID59	0000 0000_H*1	<RS-CAN0_base> + $13B0_H$	8, 16, 32
Transmit buffer pointer register 59	RSCAN0TMPTR59	0000 0000_H*1	<RS-CAN0_base> + $13B4_H$	8, 16, 32
Transmit buffer data field 0 register 59	RSCAN0TMDF059	0000 0000_H*1	<RS-CAN0_base> + $13B8_H$	8, 16, 32
Transmit buffer data field 1 register 59	RSCAN0TMDF159	0000 0000_H*1	<RS-CAN0_base> + $13BC_H$	8, 16, 32
Transmit buffer ID register 60	RSCAN0TMID60	0000 0000_H*1	<RS-CAN0_base> + $13C0_H$	8, 16, 32
Transmit buffer pointer register 60	RSCAN0TMPTR60	0000 0000_H*1	<RS-CAN0_base> + $13C4_H$	8, 16, 32
Transmit buffer data field 0 register 60	RSCAN0TMDF060	0000 0000_H*1	<RS-CAN0_base> + $13C8_H$	8, 16, 32
Transmit buffer data field 1 register 60	RSCAN0TMDF160	0000 0000_H*1	<RS-CAN0_base> + $13CC_H$	8, 16, 32
Transmit buffer ID register 61	RSCAN0TMID61	0000 0000_H*1	<RS-CAN0_base> + $13D0_H$	8, 16, 32
Transmit buffer pointer register 61	RSCAN0TMPTR61	0000 0000_H*1	<RS-CAN0_base> + $13D4_H$	8, 16, 32
Transmit buffer data field 0 register 61	RSCAN0TMDF061	0000 0000_H*1	<RS-CAN0_base> + $13D8_H$	8, 16, 32
Transmit buffer data field 1 register 61	RSCAN0TMDF161	0000 0000_H*1	<RS-CAN0_base> + $13DC_H$	8, 16, 32
Transmit buffer ID register 62	RSCAN0TMID62	0000 0000_H*1	<RS-CAN0_base> + $13E0_H$	8, 16, 32
Transmit buffer pointer register 62	RSCAN0TMPTR62	0000 0000_H*1	<RS-CAN0_base> + $13E4_H$	8, 16, 32
Transmit buffer data field 0 register 62	RSCAN0TMDF062	0000 0000_H*1	<RS-CAN0_base> + $13E8_H$	8, 16, 32
Transmit buffer data field 1 register 62	RSCAN0TMDF162	0000 0000_H*1	<RS-CAN0_base> + $13EC_H$	8, 16, 32
Transmit buffer ID register 63	RSCAN0TMID63	0000 0000_H*1	<RS-CAN0_base> + $13F0_H$	8, 16, 32
Transmit buffer pointer register 63	RSCAN0TMPTR63	0000 0000_H*1	<RS-CAN0_base> + $13F4_H$	8, 16, 32
Transmit buffer data field 0 register 63	RSCAN0TMDF063	0000 0000_H*1	<RS-CAN0_base> + $13F8_H$	8, 16, 32
Transmit buffer data field 1 register 63	RSCAN0TMDF163	0000 0000_H*1	<RS-CAN0_base> + $13FC_H$	8, 16, 32
TX history list access register 0	RSCAN0THLACC0	0000 0000_H*1	<RS-CAN0_base> + 1800_H	8, 16, 32
TX history list access register 1	RSCAN0THLACC1	0000 0000_H*1	<RS-CAN0_base> + 1804_H	8, 16, 32
TX history list access register 2	RSCAN0THLACC2	0000 0000_H*1	<RS-CAN0_base> + 1808_H	8, 16, 32
TX history list access register 3	RSCAN0THLACC3	0000 0000_H*1	<RS-CAN0_base> + $180C_H$	8, 16, 32
RAM test page access register 0	RSCAN0RPGACC0	0000 0000_H*1	<RS-CAN0_base> + 1900_H	8, 16, 32
RAM test page access register 1	RSCAN0RPGACC1	0000 0000_H*1	<RS-CAN0_base> + 1904_H	8, 16, 32
RAM test page access register 2	RSCAN0RPGACC2	0000 0000_H*1	<RS-CAN0_base> + 1908_H	8, 16, 32
RAM test page access register 3	RSCAN0RPGACC3	0000 0000_H*1	<RS-CAN0_base> + $190C_H$	8, 16, 32
RAM test page access register 4	RSCAN0RPGACC4	0000 0000_H*1	<RS-CAN0_base> + 1910_H	8, 16, 32
RAM test page access register 5	RSCAN0RPGACC5	0000 0000_H*1	<RS-CAN0_base> + 1914_H	8, 16, 32

Table 16.10 List of RS-CAN Module Registers (20/21)

Register Name	Symbol	Value after Reset	Address	Access Size
RAM test page access register 6	RSCAN0RPGACC6	0000 0000_H*1	<RS-CAN0_base> + 1918_H	8, 16, 32
RAM test page access register 7	RSCAN0RPGACC7	0000 0000_H*1	<RS-CAN0_base> + $191C_H$	8, 16, 32
RAM test page access register 8	RSCAN0RPGACC8	0000 0000_H*1	<RS-CAN0_base> + 1920_H	8, 16, 32
RAM test page access register 9	RSCAN0RPGACC9	0000 0000_H*1	<RS-CAN0_base> + 1924_H	8, 16, 32
RAM test page access register 10	RSCAN0RPGACC10	0000 0000_H*1	<RS-CAN0_base> + 1928_H	8, 16, 32
RAM test page access register 11	RSCAN0RPGACC11	0000 0000_H*1	<RS-CAN0_base> + $192C_H$	8, 16, 32
RAM test page access register 12	RSCAN0RPGACC12	0000 0000_H*1	<RS-CAN0_base> + 1930_H	8, 16, 32
RAM test page access register 13	RSCAN0RPGACC13	0000 0000_H*1	<RS-CAN0_base> + 1934_H	8, 16, 32
RAM test page access register 14	RSCAN0RPGACC14	0000 0000_H*1	<RS-CAN0_base> + 1938_H	8, 16, 32
RAM test page access register 15	RSCAN0RPGACC15	0000 0000_H*1	<RS-CAN0_base> + $193C_H$	8, 16, 32
RAM test page access register 16	RSCAN0RPGACC16	0000 0000_H*1	<RS-CAN0_base> + 1940_H	8, 16, 32
RAM test page access register 17	RSCAN0RPGACC17	0000 0000_H*1	<RS-CAN0_base> + 1944_H	8, 16, 32
RAM test page access register 18	RSCAN0RPGACC18	0000 0000_H*1	<RS-CAN0_base> + 1948_H	8, 16, 32
RAM test page access register 19	RSCAN0RPGACC19	0000 0000_H*1	<RS-CAN0_base> + $194C_H$	8, 16, 32
RAM test page access register 20	RSCAN0RPGACC20	0000 0000_H*1	<RS-CAN0_base> + 1950_H	8, 16, 32
RAM test page access register 21	RSCAN0RPGACC21	0000 0000_H*1	<RS-CAN0_base> + 1954_H	8, 16, 32
RAM test page access register 22	RSCAN0RPGACC22	0000 0000_H*1	<RS-CAN0_base> + 1958_H	8, 16, 32
RAM test page access register 23	RSCAN0RPGACC23	0000 0000_H*1	<RS-CAN0_base> + $195C_H$	8, 16, 32
RAM test page access register 24	RSCAN0RPGACC24	0000 0000_H*1	<RS-CAN0_base> + 1960_H	8, 16, 32
RAM test page access register 25	RSCAN0RPGACC25	0000 0000_H*1	<RS-CAN0_base> + 1964_H	8, 16, 32
RAM test page access register 26	RSCAN0RPGACC26	0000 0000_H*1	<RS-CAN0_base> + 1968_H	8, 16, 32
RAM test page access register 27	RSCAN0RPGACC27	0000 0000_H*1	<RS-CAN0_base> + $196C_H$	8, 16, 32
RAM test page access register 28	RSCAN0RPGACC28	0000 0000_H*1	<RS-CAN0_base> + 1970_H	8, 16, 32
RAM test page access register 29	RSCAN0RPGACC29	0000 0000_H*1	<RS-CAN0_base> + 1974_H	8, 16, 32
RAM test page access register 30	RSCAN0RPGACC30	0000 0000_H*1	<RS-CAN0_base> + 1978_H	8, 16, 32
RAM test page access register 31	RSCAN0RPGACC31	0000 0000_H*1	<RS-CAN0_base> + $197C_H$	8, 16, 32
RAM test page access register 32	RSCAN0RPGACC32	0000 0000_H*1	<RS-CAN0_base> + 1980_H	8, 16, 32
RAM test page access register 33	RSCAN0RPGACC33	0000 0000_H*1	<RS-CAN0_base> + 1984_H	8, 16, 32
RAM test page access register 34	RSCAN0RPGACC34	0000 0000_H*1	<RS-CAN0_base> + 1988_H	8, 16, 32
RAM test page access register 35	RSCAN0RPGACC35	0000 0000_H*1	<RS-CAN0_base> + $198C_H$	8, 16, 32
RAM test page access register 36	RSCAN0RPGACC36	0000 0000_H*1	<RS-CAN0_base> + 1990_H	8, 16, 32
RAM test page access register 37	RSCAN0RPGACC37	0000 0000_H*1	<RS-CAN0_base> + 1994_H	8, 16, 32
RAM test page access register 38	RSCAN0RPGACC38	0000 0000_H*1	<RS-CAN0_base> + 1998_H	8, 16, 32
RAM test page access register 39	RSCAN0RPGACC39	0000 0000_H*1	<RS-CAN0_base> + $199C_H$	8, 16, 32
RAM test page access register 40	RSCAN0RPGACC40	0000 0000_H*1	<RS-CAN0_base> + $19A0_H$	8, 16, 32
RAM test page access register 41	RSCAN0RPGACC41	0000 0000_H*1	<RS-CAN0_base> + $19A4_H$	8, 16, 32
RAM test page access register 42	RSCAN0RPGACC42	0000 0000_H*1	<RS-CAN0_base> + $19A8_H$	8, 16, 32
RAM test page access register 43	RSCAN0RPGACC43	0000 0000_H*1	<RS-CAN0_base> + $19AC_H$	8, 16, 32
RAM test page access register 44	RSCAN0RPGACC44	0000 0000_H*1	<RS-CAN0_base> + $19B0_H$	8, 16, 32
RAM test page access register 45	RSCAN0RPGACC45	0000 0000_H*1	<RS-CAN0_base> + $19B4_H$	8, 16, 32
RAM test page access register 46	RSCAN0RPGACC46	0000 0000_H*1	<RS-CAN0_base> + $19B8_H$	8, 16, 32
RAM test page access register 47	RSCAN0RPGACC47	0000 0000_H*1	<RS-CAN0_base> + $19BC_H$	8, 16, 32
RAM test page access register 48	RSCAN0RPGACC48	0000 0000_H*1	<RS-CAN0_base> + $19C0_H$	8, 16, 32
RAM test page access register 49	RSCAN0RPGACC49	0000 0000_H*1	<RS-CAN0_base> + $19C4_H$	8, 16, 32
RAM test page access register 50	RSCAN0RPGACC50	0000 0000_H*1	<RS-CAN0_base> + $19C8_H$	8, 16, 32
RAM test page access register 51	RSCAN0RPGACC51	0000 0000_H*1	<RS-CAN0_base> + $19CC_H$	8, 16, 32
RAM test page access register 52	RSCAN0RPGACC52	0000 0000_H*1	<RS-CAN0_base> + $19D0_H$	8, 16, 32
RAM test page access register 53	RSCAN0RPGACC53	0000 0000_H*1	<RS-CAN0_base> + $19D4_H$	8, 16, 32
RAM test page access register 54	RSCAN0RPGACC54	0000 0000_H*1	<RS-CAN0_base> + $19D8_H$	8, 16, 32

Table 16.10 List of RS-CAN Module Registers (21/21)

Register Name	Symbol	Value after Reset	Address	Access Size
RAM test page access register 55	RSCAN0RPGACC55	0000 0000_H*1	<RS-CAN0_base> + $19DC_H$	8, 16, 32
RAM test page access register 56	RSCAN0RPGACC56	0000 0000_H*1	<RS-CAN0_base> + $19E0_H$	8, 16, 32
RAM test page access register 57	RSCAN0RPGACC57	0000 0000_H*1	<RS-CAN0_base> + $19E4_H$	8, 16, 32
RAM test page access register 58	RSCAN0RPGACC58	0000 0000_H*1	<RS-CAN0_base> + $19E8_H$	8, 16, 32
RAM test page access register 59	RSCAN0RPGACC59	0000 0000_H*1	<RS-CAN0_base> + $19EC_H$	8, 16, 32
RAM test page access register 60	RSCAN0RPGACC60	0000 0000_H*1	<RS-CAN0_base> + $19F0_H$	8, 16, 32
RAM test page access register 61	RSCAN0RPGACC61	0000 0000_H*1	<RS-CAN0_base> + $19F4_H$	8, 16, 32
RAM test page access register 62	RSCAN0RPGACC62	0000 0000_H*1	<RS-CAN0_base> + $19F8_H$	8, 16, 32
RAM test page access register 63	RSCAN0RPGACC63	0000 0000_H*1	<RS-CAN0_base> + $19FC_H$	8, 16, 32

Note 1. The actual registers are not memory-mapped but connected to the corresponding locations in RAM listed below, so the values are those following initialization of the RAM rather than the actual values of the registers following a reset.

Table 16.11 Transmit Buffer p Assigned for Each Channel

	CANm
Transmit buffer p	Transmit buffer 16 × m + 0
	Transmit buffer 16 × m + 1
	Transmit buffer 16 × m + 2
	Transmit buffer 16 × m + 3
	Transmit buffer 16 × m + 4
	Transmit buffer 16 × m + 5
	Transmit buffer 16 × m + 6
	Transmit buffer 16 × m + 7
	Transmit buffer 16 × m + 8
	Transmit buffer 16 × m + 9
	Transmit buffer 16 × m + 10
	Transmit buffer 16 × m + 11
	Transmit buffer 16 × m + 12
	Transmit buffer 16 × m + 13
	Transmit buffer 16 × m + 14
	Transmit buffer 16 × m + 15

Table 16.12 Transmit/Receive FIFO Buffer k Assigned for Each Channel

	CANm
Transmit/receive FIFO buffer k	Transmit/receive FIFO buffer 3 × m + 0
	Transmit/receive FIFO buffer 3 × m + 1
	Transmit/receive FIFO buffer 3 × m + 2

Table 16.13 Transmit Buffer p Linked to Transmit/Receive FIFO Buffer by the Set Value of CFTML[3:0] Bits

Set value of CFTML[3:0] bit	Transmit buffer p linked to transmit/receive FIFO buffer
0000_B	Transmit buffer 16 × m + 0
0001_B	Transmit buffer 16 × m + 1
0010_B	Transmit buffer 16 × m + 2
0011_B	Transmit buffer 16 × m + 3
0100_B	Transmit buffer 16 × m + 4
0101_B	Transmit buffer 16 × m + 5
0110_B	Transmit buffer 16 × m + 6
0111_B	Transmit buffer 16 × m + 7
1000_B	Transmit buffer 16 × m + 8
1001_B	Transmit buffer 16 × m + 9
1010_B	Transmit buffer 16 × m + 10
1011_B	Transmit buffer 16 × m + 11
1100_B	Transmit buffer 16 × m + 12
1101_B	Transmit buffer 16 × m + 13
1110_B	Transmit buffer 16 × m + 14
1111_B	Transmit buffer 16 × m + 15

Table 16.14 Transmit Buffer p Assigned for Transmit Queue for Each Channel

Set Value of TXQDC[3:0] Bits	Transmit Buffer p Assigned for Transmit Queue
0000_B	Should not be set.
0001_B	Should not be set.
0010_B	Transmit buffer 16 × m + 15 to 16 × m + 13
0011_B	Transmit buffer 16 × m + 15 to 16 × m + 12
0100_B	Transmit buffer 16 × m + 15 to 16 × m + 11
0101_B	Transmit buffer 16 × m + 15 to 16 × m + 10
0110_B	Transmit buffer 16 × m + 15 to 16 × m + 9
0111_B	Transmit buffer 16 × m + 15 to 16 × m + 8
1000_B	Transmit buffer 16 × m + 15 to 16 × m + 7
1001_B	Transmit buffer 16 × m + 15 to 16 × m + 6
1010_B	Transmit buffer 16 × m + 15 to 16 × m + 5
1011_B	Transmit buffer 16 × m + 15 to 16 × m + 4
1100_B	Transmit buffer 16 × m + 15 to 16 × m + 3
1101_B	Transmit buffer 16 × m + 15 to 16 × m + 2
1110_B	Transmit buffer 16 × m + 15 to 16 × m + 1
1111_B	Transmit buffer 16 × m + 15 to 16 × m + 0

16.3.2 Register Details

16.3.2.1 RSCAN0CmCFG — Channel Configuration Register (m = 0 to 3)

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: <RS-CAN0_base> + 0000_H + (m × 0010_H)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	SJW[1:0]		—	TSEG2[2:0]			TSEG1[3:0]			
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	BRP[9:0]									
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 16.15 RSCAN0CmCFG Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 26	—	Reserved These bits are always read as 0. When writing, write 0.
25, 24	SJW [1:0]	Resynchronization Jump Width Control b25 b24 0 0: 1 Tq 0 1: 2 Tq 1 0: 3 Tq 1 1: 4 Tq
23	—	Reserved This bit is always read as 0. When writing, write 0.
22 to 20	TSEG2 [2:0]	Time Segment 2 Control b22 b21 b20 0 0 0: Setting prohibited 0 0 1: 2 Tq 0 1 0: 3 Tq 0 1 1: 4 Tq 1 0 0: 5 Tq 1 0 1: 6 Tq 1 1 0: 7 Tq 1 1 1: 8 Tq

Table 16.15 RSCAN0CmCFG Register Contents (2/2)

Bit Position	Bit Name	Function
19 to 16	TSEG1 [3:0]	Time Segment 1 Control b19 b18 b17 b16 0 0 0 0: Setting prohibited 0 0 0 1: Setting prohibited 0 0 1 0: Setting prohibited 0 0 1 1: 4Tq 0 1 0 0: 5Tq 0 1 0 1: 6Tq 0 1 1 0: 7Tq 0 1 1 1: 8Tq 1 0 0 0: 9Tq 1 0 0 1: 10Tq 1 0 1 0: 11Tq 1 0 1 1: 12Tq 1 1 0 0: 13Tq 1 1 0 1: 14Tq 1 1 1 0: 15 Tq 1 1 1 1: 16 Tq
15 to 10	—	Reserved These bits are always read as 0. When writing, write 0.
9 to 0	BRP [9:0]	Prescaler Division Ratio Set When these bits are set to P (0 to 1023), the baud rate prescaler divides fCAN by P+1.

Modify the RSCAN0CmCFG register in channel reset mode or channel halt mode. Set this register before making a transition to channel communication mode. For setting bit timing, see **Section 16.5.1, Initial Settings**.

SJW[1:0] Bits

These bits are used to specify a Tq value for the resynchronization jump width. A value of 1 Tq to 4 Tq can be set. Set a value equal to or smaller than the value of the TSEG2 bits.

TSEG2[2:0] Bits

These bits are used to specify a Tq value for the length of phase buffer segment 2 (PHASE_SEG2).

A value of 2 Tq to 8 Tq can be set.

Set a value smaller than the value of the TSEG1 bits.

TSEG1[3:0] Bits

These bits are used to specify a Tq value for the total length of the propagation time segment (PROP_SEG) and phase buffer segment 1 (PHASE_SEG1).

A value of 4 Tq to 16 Tq can be set.

BRP[9:0] Bits

The CANmTq clock (fCANTQm) is obtained by dividing the CAN clock (fCAN) by the baud rate prescaler ((BRP[9:0] bits + 1). One clock cycle of the CANmTq clock is 1 time quantum (Tq).

16.3.2.2 RSCAN0CmCTR — Channel Control Register (m = 0 to 3)

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: <RS-CAN0_base> + 0004_H + (m × 0010_H)

Value after reset: 0000 0005_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	CTMS[1:0]		CTME	ERRD	BOM[1:0]		—	—	—	—	TAIE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ALIE	BLIE	OLIE	BORIE	BOEIE	EPIE	EWIE	BEIE	—	—	—	—	RTBO	CSLPR	CHMDC[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 16.16 RSCAN0CmCTR Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 27	—	Reserved These bits are always read as 0. When writing, write 0.
26, 25	CTMS[1:0]	Communication Test Mode Select b26 b25 0 0 : Standard test mode 0 1 : Listen-only mode 1 0 : Self-test mode 0 (external loopback mode) 1 1 : Self-test mode 1 (internal loopback mode)
24	CTME	Communication Test Mode Enable 0: Communication test mode is disabled. 1: Communication test mode is enabled.
23	ERRD	Error Display Mode Select 0: Only the first error information is displayed after bit 14 to 8 in this register are all cleared. 1: All error information is displayed.
22, 21	BOM[1:0]	Bus Off Recovery Mode Select b22 b21 0 0 : ISO11898-1 compliant 0 1 : Transition to channel halt mode at bus-off entry 1 0 : Transition to channel halt mode at bus-off end 1 1 : Transition to channel halt mode (during bus-off recovery period) by a program request
20 to 17	—	Reserved These bits are always read as 0. When writing, write 0.
16	TAIE	Transmit Abort Interrupt Enable 0: Transmit abort interrupt is disabled. 1: Transmit abort interrupt is enabled.
15	ALIE	Arbitration Lost Interrupt Enable 0: Arbitration lost interrupt is disabled. 1: Arbitration lost interrupt is enabled.
14	BLIE	Bus Lock Interrupt Enable 0: Bus lock interrupt is disabled. 1: Bus lock interrupt is enabled.
13	OLIE	Overload Frame Transmission Interrupt Enable 0: Overload frame transmission interrupt is disabled. 1: Overload frame transmission interrupt is enabled.

Table 16.16 RSCAN0CmCTR Register Contents (2/2)

Bit Position	Bit Name	Function
12	BORIE	Bus Off Recovery Interrupt Enable 0: Bus off recovery interrupt is disabled. 1: Bus off recovery interrupt is enabled.
11	BOEIE	Bus Off Entry Interrupt Enable 0: Bus off entry interrupt is disabled. 1: Bus off entry interrupt is enabled.
10	EPIE	Error Passive Interrupt Enable 0: Error passive interrupt is disabled. 1: Error passive interrupt is enabled.
9	EWIE	Error Warning Interrupt Enable 0: Error warning interrupt is disabled. 1: Error warning interrupt is enabled.
8	BEIE	Bus Error Interrupt Enable 0: Bus error interrupt is disabled. 1: Bus error interrupt is enabled.
7 to 4	—	Reserved These bits are always read as 0. When writing, write 0.
3	RTBO	Forcible Return from Bus-off When this bit is set to 1, forcible return from the bus off state is made. This bit is always read as 0.
2	CSLPR	Channel Stop Mode 0: Other than channel stop mode 1: Channel stop mode
1,0	CHMDC[1:0]	Mode Select b1 b0 0 0: Channel communication mode 0 1: Channel reset mode 1 0: Channel halt mode 1 1: Setting prohibited

CTMS[1:0] Bits

These bits are used to select a communication test mode. Modify these bits in channel halt mode. These bits are set to 0 in channel reset mode.

CTME Bit

Setting this bit to 1 enables communication test mode. Modify this bit in channel halt mode. This bit is set to 0 in channel reset mode.

ERRD Bit

This bit is used to control display mode of bits 14 to 8 in the RSCAN0CmERFL register.

When this bit is cleared to 0, only the flags of the first error are set to 1. If two or more errors occur first, all the flags of detected errors are set to 1.

When this bit is set to 1, all the flags of errors that have occurred are set to 1 regardless of the error occurrence order.

Modify this bit in channel reset mode or channel halt mode.

BOM[1:0] Bits

These bits are used to select a bus off recovery mode of the RS-CAN module.

When the BOM[1:0] bits are set to 00_B, return from the bus off state is compliant with the CAN specifications. That is, the RS-CAN module reenters the CAN communication (error active state) after 11 consecutive recessive bits are detected 128 times. A bus off recovery interrupt request is generated at the time of return from the bus off state. If the CHMDC[1:0] bits are set to 10_B (channel halt mode) before recessive bits are detected 128 times, the RS-CAN module does not transition to channel halt mode until recessive bits are detected 128 times.

When the RS-CAN module reaches the bus off state when the BOM[1:0] bits are set to 01_B, the CHMDC[1:0] bits in the RSCAN0CmCTR register (m = 0 to 3) are set to 10_B and the RS-CAN module transitions to channel halt mode. No bus off recovery interrupt request is generated at the time of return from the bus off state and the TEC[7:0] and REC[7:0] bits in the RSCAN0CmSTS register are cleared to 00_H.

When the RS-CAN module reaches the bus off state when the BOM[1:0] bits are set to 10_B, the CHMDC[1:0] bits are set to 10_B and the RS-CAN module transitions to channel halt mode after return from the bus off state (11 consecutive recessive bits are detected 128 times). A bus off recovery interrupt request is generated at the time of return from the bus off state and the TEC[7:0] and REC[7:0] bits are cleared to 00_H.

When the BOM[1:0] bits are set to 11_B and the CHMDC[1:0] bits are set to 10_B while the RS-CAN module is in the bus off state, the RS-CAN module transitions to channel halt mode. No bus off recovery interrupt request is generated at the time of return from the bus off state and the TEC[7:0] and REC[7:0] bits are cleared to 00_H. However, if 11 consecutive recessive bits are detected 128 times and the RS-CAN module has recovered to the error active state from the bus off state before the CHMDC[1:0] bits are set to 10_B, a bus off recovery interrupt request is generated.

When the RS-CAN module transitions to channel halt mode (at bus off entry when the BOM[1:0] bits are 01_B or at bus off end when the BOM[1:0] bits are 10_B), if the CHMDC[1:0] bits are written by a program at the same time, writing by a program takes precedence. Modify BOM[1:0] bits in channel reset mode.

TAIE Bit

When transmit abort of the transmit buffer is completed with the TAIE bit set to 1, an interrupt request is generated. Modify this bit in channel reset mode.

ALIE Bit

When the ALF flag in the RSCAN0CmERFL register is set to 1 with the ALIE bit set to 1, an error interrupt request is generated. Modify this bit in channel reset mode.

BLIE Bit

When the BLF flag in the RSCAN0CmERFL register is set to 1 with the BLIE bit set to 1, an error interrupt request is generated. Modify this bit in channel reset mode.

OLIE Bit

When the OVLF flag in the RSCAN0CmERFL register is set to 1 with the OLIE bit set to 1, an error interrupt request is generated. Modify this bit in channel reset mode.

BORIE Bit

When the BORF flag in the RSCAN0CmERFL register is set to 1 with the BORIE bit set to 1, an error interrupt request is generated. Modify this bit in channel reset mode.

BOEIE Bit

When the BOEF flag in the RSCAN0CmERFL register is set to 1 with the BOEIE bit set to 1, an error interrupt request is generated. Modify this bit in channel reset mode.

EPIE Bit

When the EPF flag in the RSCAN0CmERFL register is set to 1 with the EPIE bit set to 1, an error interrupt request is generated. Modify this bit in channel reset mode.

EWIE Bit

When the EWF flag in the RSCAN0CmERFL register is set to 1 with the EWIE bit set to 1, an error interrupt request is generated. Modify this bit in channel reset mode.

BEIE Bit

When the BEF flag in the RSCAN0CmERFL register is set to 1 with the BEIE bit set to 1, an error interrupt request is generated. Modify this bit in channel reset mode.

RTBO Bit

Setting this bit to 1 (forcible return from the bus off state) in the bus off state returns the state forcibly from the bus off state to the error active state. This bit is automatically cleared to 0. Setting this bit to 1 clears the TEC[7:0] and REC[7:0] bits in the RSCAN0CmSTS register to 00_H and also clears the BOSTS flag in the RSCAN0CmSTS register to 0 (not in bus off state). The other registers remain unchanged. No bus off recovery interrupt request due to return from the bus off state is generated. Use this bit only when the BOM[1:0] bits in the RSCAN0CmCTR register are 00_B (ISO11898-1 compliant).

A delay of up to 1 CAN bit time occurs after the RTBO bit is set to 1 until the RS-CAN module transitions to the error active state. Set this bit to 1 in channel communication mode.

CSLPR Bit

Setting this bit to 1 places the channel in channel stop mode.

Clearing this bit to 0 makes the channel leave from channel stop mode.

Do not rewrite this bit in channel communication mode or in channel halt mode.

CHMDC[1:0] Bits

These bits are used to select a channel mode (channel communication mode, channel reset mode, or channel halt mode). For details, see **Section 16.4.2.6, Channel Modes**. Setting the CSLPR bit to 1 in channel reset mode allows transition to channel stop mode. Do not set these bits to 11_B. When the RS-CAN module has transitioned to channel halt mode by the setting of the BOM[1:0] bits, the CHMDC[1:0] bits automatically becomes 10_B.

16.3.2.3 RSCAN0CmSTS — Channel Status Register (m = 0 to 3)

Access: This register can be read in 8-, 16-, or 32-bit units.

Address: <RS-CAN0_base> + 0008_H + (m × 0010_H)

Value after reset: 0000 0005_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TEC[7:0]								REC[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	COM STS	REC STS	TRM STS	BOSTS	EPSTS	CSLP STS	CHLT STS	CRST STS
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 16.17 RSCAN0CmSTS Register Contents

Bit Position	Bit Name	Function
31 to 24	TEC [7:0]	The transmit error counter (TEC) can be read.
23 to 16	REC [7:0]	The receive error counter (REC) can be read.
15 to 8	—	Reserved These bits are always read as 0.
7	COMSTS	Communication Status Flag 0: Communication is not ready. 1: Communication is ready.
6	RECSTS	Receive Status Flag 0: Bus idle, in transmission, or bus off state 1: In reception
5	TRMSTS	Transmit Status Flag 0: Bus idle or in reception 1: In transmission or bus off state
4	BOSTS	Bus Off Status Flag 0: Not in bus off state 1: In bus off state
3	EPSTS	Error Passive Status Flag 0: Not in error passive state 1: In error passive state
2	CSLPSTS	Channel Stop Status Flag 0: Not in channel stop mode 1: In channel stop mode
1	CHLTSTS	Channel Halt Status Flag 0: Not in channel halt mode 1: In channel halt mode
0	CRSTSTS	Channel Reset Status Flag 0: Not in channel reset mode 1: In channel reset mode

TEC[7:0] Bits

These bits indicate the transmit error counter value. For transmit error counter increment/decrement conditions, see the *CAN specification (ISO11898-1)*.

These bits are cleared to 0 in channel reset mode.

REC[7:0] Bits

These bits indicate the receive error counter value. For receive error counter increment/decrement conditions, see the *CAN specification (ISO11898-1)*.

These bits are cleared to 0 in channel reset mode.

COMSTS Flag

This flag indicates that communication is ready.

This flag becomes 1 when the RS-CAN module has detected 11 consecutive recessive bits after it has transitioned from channel reset mode or channel halt mode to channel communication mode. This flag is cleared to 0 in channel reset mode or channel halt mode.

RECSTS Flag

This flag is set to 1 when reception has started, and is cleared to 0 when the bus has become idle or transmission has started.

TRMSTS Flag

This flag is set to 1 when transmission has started, and is cleared to 0 when the bus has become idle or reception has started. This flag remains 1 in the bus off state.

BOSTS Flag

This flag is set to 1 when the TEC[7:0] value exceeds 255 and the RS-CAN module has entered the bus off state (TEC[7:0] value > 255), and is cleared to 0 when the RS-CAN module has exited the bus off state.

EPSTS Flag

This flag is set to 1 when the TEC[7:0] or REC[7:0] value exceeds 127 and the RS-CAN module has entered the error passive state (128 ≤ TEC[7:0] value ≤ 255 or 128 ≤ REC[7:0] value), and is cleared to 0 when the RS-CAN module has exited the error passive state or has entered channel reset mode.

CSLPSTS Flag

This flag is set to 1 when the RS-CAN module has transitioned to channel stop mode, and is cleared to 0 when the RS-CAN module has returned from channel stop mode.

CHLTSTS Flag

This flag is set to 1 when the RS-CAN module has transitioned to channel halt mode, and is cleared to 0 when the RS-CAN module has exited channel halt mode.

CRSTSTS Flag

This flag is set to 1 when the RS-CAN module has transitioned to channel reset mode, and is cleared to 0 when the RS-CAN module has transitioned to channel communication mode or channel halt mode. This flag remains 1 if the RS-CAN module transitions from channel reset mode to channel stop mode.

16.3.2.4 RSCAN0CmERFL — Channel Error Flag Register (m = 0 to 3)

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: <RS-CAN0_base> + $000C_H$ + (m × 0010_H)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	CRCREG[14:0]														
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	ADERR	B0ERR	B1ERR	CERR	AERR	FERR	SERR	ALF	BLF	OVLF	BORF	BOEF	EPF	EWF	BEF
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1

Note 1. The only effective value for writing to this flag bit is 0, which clears the status. Any other writing to the bit results in retention of the status and does not change the value.

Table 16.18 RSCAN0CmERFL Register Contents (1/2)

Bit Position	Bit Name	Function
31	—	Reserved This bit is always read as 0. When writing, write 0.
30 to 16	CRCREG[14:0]	CRC Calculation Data A CRC value calculated based on the transmitted message or received message is indicated.
15	—	Reserved This bit is always read as 0. When writing, write 0.
14	ADERR	ACK Delimiter Error Flag 0: No ACK delimiter error is detected. 1: ACK delimiter error is detected.
13	B0ERR	Dominant Bit Error Flag 0: No dominant bit error is detected. 1: Dominant bit error is detected.
12	B1ERR	Recessive Bit Error Flag 0: No recessive bit error is detected. 1: Recessive bit error is detected.
11	CERR	CRC Error Flag 0: No CRC error is detected. 1: CRC error is detected.
10	AERR	ACK Error Flag 0: No ACK error is detected. 1: ACK error is detected.
9	FERR	Form Error Flag 0: No form error is detected. 1: Form error is detected.
8	SERR	Stuff Error Flag 0: No stuff error is detected. 1: Stuff error is detected.
7	ALF	Arbitration Lost Flag 0: No arbitration lost is detected. 1: Arbitration lost is detected.
6	BLF	Bus Lock Flag 0: No channel bus lock is detected. 1: Channel bus lock is detected.

Table 16.18 RSCAN0CmERFL Register Contents (2/2)

Bit Position	Bit Name	Function
5	OVLF	Overload Flag 0: No overload is detected. 1: Overload is detected.
4	BORF	Bus Off Recovery Flag 0: No bus off recovery is detected. 1: Bus off recovery is detected.
3	BOEF	Bus Off Entry Flag 0: No bus off entry is detected. 1: Bus off entry is detected.
2	EPF	Error Passive Flag 0: No error passive is detected. 1: Error passive is detected.
1	EWF	Error Warning Flag 0: No error warning is detected. 1: Error warning is detected.
0	BEF	Bus Error Flag 0: No channel bus error is detected. 1: Channel bus error is detected.

See the *CAN specification (ISO11898-1)* if you want to check error occurrence conditions. To clear each flag of this register, write 0 by the program. These flags cannot be set to 1 by the program. If any of these flags becomes 1 at the timing when the program writes 0 to the flag, the flag is set to 1. Each flag is cleared to 0 in channel reset mode.

As for bits 14 to 8 in the RSCAN0CmERFL register, if an error is detected with all flags of bits 14 to 8 set to 0 when the ERRD bit in the RSCAN0CmCTR register is set to 0 (only the first error information is displayed), the corresponding flag is set to 1.

CRCREG[14:0] Flags

When the CTME bit in the RSCAN0CmCTR register is set to 1 (communication test mode is enabled), the CRC value calculated based on the transmitted or received message can be read. When the CTME bit is set to 0 (communication test mode is disabled), these bits are always read as 0.

ADERR Flag

This flag is set to 1 when a form error has been detected in the ACK delimiter during transmission.

B0ERR Flag

This flag is set to 1 when a recessive bit has been detected even though a dominant bit was transmitted.

B1ERR Flag

This flag is set to 1 when a dominant bit has been detected even though a recessive bit was transmitted.

CERR Flag

This flag is set to 1 when a CRC error has been detected.

AERR Flag

This flag is set to 1 when an ACK error has been detected.

FERR Flag

This flag is set to 1 when a form error has been detected.

SERR Flag

This flag is set to 1 when a stuff error has been detected.

ALF Flag

This flag is set to 1 when an arbitration lost has been detected.

BLF Flag

This flag is set to 1 when 32 consecutive dominant bits have been detected on the CAN bus in channel communication mode. After that, the detection processing restarts at either timing below.

- A recessive bit is detected after the BLF bit has been modified from 1 to 0.
- The RS-CAN module transitions to channel reset mode and returns to channel communication mode after the BLF bit has been modified from 1 to 0.

OVLF Flag

This flag is set to 1 when the overload frame transmit condition has been detected when performing reception or transmission.

BORF Flag

This flag is set to 1 when 11 consecutive recessive bits have been detected 128 times and the RS-CAN module returns from the bus off state. However, this flag is not set to 1 if the RS-CAN module returns from the bus off state in either of the following cases before 11 consecutive recessive bits are detected 128 times.

- The CHMDC[1:0] bits in the RSCAN0CmCTR register are set to 01_B (channel reset mode).
- The RTBO bit in the RSCAN0CmCTR register is set to 1 (forcible return from the bus off state).
- The BOM[1:0] bits in the RSCAN0CmCTR register are set to 01_B (transition to channel halt mode at bus off entry).
- The CHMDC[1:0] bits in the RSCAN0CmCTR register are set to 10_B (channel halt mode) before 11 consecutive recessive bits are detected 128 times with the BOM[1:0] bits set to 11_B (transition to channel halt mode upon a request from the program during bus off).

BOEF Flag

This flag is set to 1 when the state becomes bus off state (TEC[7:0] value > 255). This flag is also set to 1 when the state becomes bus off state with the BOM[1:0] bits in the RSCAN0CmCTR register (m = 0 to 3) set to 01_B (transition to channel halt mode at bus off entry).

EPF Flag

This flag becomes 1 when the RS-CAN module becomes error passive state (REC[7:0] or TEC[7:0] value > 127). This flag becomes 1 only when the REC[7:0] or TEC[7:0] value exceeds 127 for the first time. Therefore, if the program writes 0 to this flag with the REC[7:0] or TEC[7:0] value remaining over 127, this bit is not set to 1 until both REC [7:0] and TEC[7:0] values become 127 or less and then either the REC[7:0] or TEC[7:0] value exceeds 127 again.

EWF Flag

This flag is set to 1 only when the REC[7:0] or TEC[7:0] value exceeds 95 for the first time. Therefore, if the program writes 0 to this flag with the REC[7:0] or TEC[7:0] value remaining over 95, this bit is not set to 1 until both REC [7:0] and TEC[7:0] values become 95 or less and then either the REC[7:0] or TEC[7:0] value exceeds 95 again.

BEF Flag

This flag is set to 1 when any one of the ADERR, B0ERR, B1ERR, CERR, AERR, FERR, and SERR flags in the RSCAN0CmERFL register is set to 1.

NOTE

When writing 0 to flags of this register, write 0 to corresponding flags and 1 to other flags by using a store instruction. Note that when using a bit-manipulation instruction for writing to this register, the read-modify-write operation may inadvertently clear the flags.

16.3.2.5 RSCAN0GCFG — Global Configuration Register

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: <RS-CAN0_base> + 0084_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ITRCP[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSBTCS[2:0]			TSSS	TSP[3:0]				—	—	—	DCS	MME	DRE	DCE	TPRI
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Table 16.19 RSCAN0GCFG Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 16	ITRCP[15:0]	Interval Timer Prescaler Set When these bits are set to q, the peripheral function clock A is divided by q. When using interval timer, setting 0000_H is prohibited.
15 to 13	TSBTCS[1:0]	Timestamp Clock Source Select b15 b14 b13 0 0 0 : Channel 0 bit time clock 0 0 1 : Channel 1 bit time clock 0 1 0 : Channel 2 bit time clock 0 1 1 : Channel 3 bit time clock 1 0 0 : Setting prohibited 1 0 1 : Setting prohibited 1 1 0 : Setting prohibited 1 1 1 : Setting prohibited
12	TSSS	Timestamp Source Select 0: pclk/n2*1 1: Bit time clock is used as the source clock of the timestamp counter.
11 to 8	TSP[3:0]	Timestamp Clock Source Division b11 b10 b9 b8 0 0 0 0 : Not divided 0 0 0 1 : Divided by 2 0 0 1 0 : Divided by 4 0 0 1 1 : Divided by 8 0 1 0 0 : Divided by 16 0 1 0 1 : Divided by 32 0 1 1 0 : Divided by 64 0 1 1 1 : Divided by 128 1 0 0 0 : Divided by 256 1 0 0 1 : Divided by 512 1 0 1 0 : Divided by 1024 1 0 1 1 : Divided by 2048 1 1 0 0 : Divided by 4096 1 1 0 1 : Divided by 8192 1 1 1 0 : Divided by 16384 1 1 1 1 : Divided by 32768
7 to 5	—	Reserved These bits are always read as 0. When writing, write 0.
4	DCS	CAN Clock Source Select 0: clkc 1: clk_xincan*2

Table 16.19 RSCAN0GCFG Register Contents (2/2)

Bit Position	Bit Name	Function
3	MME	Mirror Function Enable 0: Mirror function is disabled. 1: Mirror function is enabled.
2	DRE	DLC Replacement Enable 0: DLC replacement is disabled. 1: DLC replacement is enabled.
1	DCE	DLC Check Enable 0: DLC check is disabled. 1: DLC check is enabled.
0	TPRI	Transmit Priority Select 0: ID priority 1: Transmit buffer number priority

Note 1. Set TSBTCS[2:0] to 000_B before select pclk/2 as the timestamp count source.
Note 2. Set the CAN clock frequency to be pclk/2 or below.

Modify the RSCAN0GCFG register in global reset mode.

ITRCP[15:0] Bits

These bits are used to set a clock source division value of the interval timer for FIFO buffers. For details, see **Section 16.4.4.6, Interval Transmission Function**.

TSBTCS[2:0] Bits

When the TSSS bit is 1, these bits are used to select the channel of the bit time clock that will be the clock source of the timestamp counter.

TSSS Bit

This bit is used to select a clock source of the timestamp counter.

TSP[3:0] Bits

These bits specify the prescaler used to divide the clock source (selected by the TSBTCS[2:0] and TSSS bits) to generate the timestamp counter clock signal.

DCS Bit

When this bit is set to 0, clkc is used as the clock source of the CAN clock (fCAN).

When this bit is set to 1, the main clock is used as the clock source of the CAN clock.

Set the frequency of CAN clock (fCAN) at 1/2 of pclk.

MME Bit

Setting this bit to 1 makes the mirror function available.

DRE Bit

When the DRE bit is set to 1, the DLC value of the receive rule is stored in the buffer instead of the DLC value of the received message after the DLC value has passed through the DLC filter. In this case, a value of 00_H is stored in the data byte that exceeds the DLC value of the receive rule.

When the DCE bit is set to 1 (DLC check is enabled), the DLC replacement function is available.

DCE Bit

Setting this bit to 1 makes the DLC check function available. Set the GAFLDLC[3:0] bits in the RSCAN0GAFLP0j register to 0000_B before clearing the DCE bit in the RSCAN0GCFG register to 0.

TPRI Bit

This bit is used to set the transmit priority.

When this bit is set to 0, ID priority is selected and the transmit priority complies with the CAN bus arbitration rule (ISO11898-1 specifications). When this bit is set to 1, transmit buffer number priority is selected and the lowest-numbered transmit buffer among those specified for transmission takes precedence.

While the transmit queue is in use, clear this bit to 0.

16.3.2.6 RSCAN0GCTR — Global Control Register

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: <RS-CAN0_base> + 0088_H

Value after reset: $0000\ 0005_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	TSRST
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	THLEIE	MEIE	DEIE	—	—	—	—	—	GSLPR	GMDC[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1
R/W	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W

Table 16.20 RSCAN0GCTR Register Contents

Bit Position	Bit Name	Function
31 to 17	—	Reserved These bits are always read as 0. When writing, write 0.
16	TSRST	Timestamp Counter Reset Setting the TSRST bit to 1 resets the timestamp counter. This bit is always read as 0.
15 to 11	—	Reserved These bits are always read as 0. When writing, write 0.
10	THLEIE	Transmit History Buffer Overflow Interrupt Enable 0: Transmit history buffer overflow interrupt is disabled. 1: Transmit history buffer overflow interrupt is enabled.
9	MEIE	FIFO Message Lost Interrupt 0: FIFO message lost interrupt is disabled. 1: FIFO message lost interrupt is enabled.
8	DEIE	DLC Error Interrupt Enable 0: DLC error interrupt is disabled. 1: DLC error interrupt is enabled.
7 to 3	—	Reserved These bits are always read as 0. When writing, write 0.
2	GSLPR	Global Stop Mode 0: Other than global stop mode 1: Global stop mode
1, 0	GMDC[1:0]	Global Mode Select b1 b0 0 0: Global operating mode 0 1: Global reset mode 1 0: Global test mode 1 1: Setting prohibited

TSRST Bit

This bit is used to reset the timestamp counter. When this bit is set to 1, the RSCAN0GTSC register is cleared to 0000_H.

THLEIE Bit

When the THLEIE bit is set to 1 and the THLES flag in the RSCAN0GERFL register is set to 1, an interrupt request is generated. Modify this bit in global reset mode.

MEIE Bit

When the MEIE bit is set to 1 and the MES flag in the RSCAN0GERFL register is set to 1, an interrupt request is generated. Modify this bit in global reset mode.

DEIE Bit

When the DEIE bit is set to 1 and the DEF flag in the RSCAN0GERFL register is set to 1, an interrupt request is generated. Modify this bit in global reset mode.

GSLPR Bit

Setting this bit to 1 places the RS-CAN module in global stop mode.

Clearing this bit to 0 makes the RS-CAN module leave from global stop mode.

Do not modify this bit from 0 to 1 in global operation mode or global test mode.

GMDC[1:0] Bits

These bits are used to select a mode of the entire RS-CAN module (global operating mode, global reset mode, or global test mode). For details, see **Section 16.4.2.1, Global Modes**. Setting the GSLPR bit to 1 in global reset mode makes the RS-CAN module transition to global stop mode.

16.3.2.7 RSCAN0GSTS — Global Status Register

Access: This register can be read in 8-, 16-, or 32-bit units.

Address: <RS-CAN0_base> + $008C_H$

Value after reset: 0000 $000D_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	GRAM INIT	GSLP STS	GHLT STS	GRST STS
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 16.21 RSCAN0GSTS Register Contents

Bit Position	Bit Name	Function
31 to 4	—	Reserved These bits are always read as 0.
3	GRAMINIT	CAN RAM Initialization Status Flag 0: CAN RAM initialization is completed. 1: CAN RAM initialization is ongoing.
2	GSLPSTS	Global Stop Status Flag 0: Not in global stop mode 1: In global stop mode
1	GHLTSTS	Global Test Status Flag 0: Not in global test mode 1: In global test mode
0	GRSTSTS	Global Reset Status Flag 0: Not in global reset mode 1: In global reset mode

GRAMINIT Flag

This flag indicates the initialization status of the CAN RAM.

This flag is set to 1 after the MCU has been reset, and is cleared to 0 when CAN RAM initialization is completed.

GSLPSTS Flag

This flag is set to 1 when the RS-CAN module has transitioned to global stop mode, and is cleared to 0 when the RS-CAN module has returned from global stop mode.

GHLTSTS Flag

This flag is set to 1 when the RS-CAN module has transitioned to global test mode, and is cleared to 0 when the RS-CAN module has exited global test mode.

GRSTSTS Flag

This flag is set to 1 when the RS-CAN module has transitioned to global reset mode, and is cleared to 0 when the RS-CAN module has exited global reset mode. This flag remains 1 if the RS-CAN module has transitioned from global reset mode to global stop mode.

16.3.2.8 RSCAN0GERFL — Global Error Flag Register

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: <RS-CAN0_base> + 0090_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	THLES	MES	DEF
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W*1

Note 1. The only effective value for writing to this flag bit is 0, which clears the status. Any other writing to the bit results in retention of the status and does not change the value.

Table 16.22 RSCAN0GERFL Register Contents

Bit Position	Bit Name	Function
31 to 3	—	Reserved The read value is undefined. When writing, write 0.
2	THLES	Transmit History Buffer Overflow Status Flag 0: No transmit history buffer overflow is present. 1: A transmit history buffer overflow is present.
1	MES	FIFO Message Lost Status Flag 0: No FIFO message lost error is present. 1: A FIFO message lost error is present.
0	DEF	DLC Error Flag 0: No DLC error is present. 1: A DLC error is present.

All flags in the RSCAN0GERFL register are cleared to 0 in global reset mode.

THLES Flag

The THLES flag is set to 1 when any one of the THLELT flags in the RSCAN0THLSTSm register (m = 0 to 3) is set to 1.

This flag is cleared to 0 when the THLELT flags of all channels are set to 0.

MES Flag

The MES flag is set to 1 when any one of the RFMLT flags in the RSCAN0RFSTSx register (x = 0 to 7) or the CFMLT flags in the RSCAN0CFSTSk register (k = 0 to 11) is set to 1.

This flag is cleared to 0 when all the RFMLT flags and CFMLT flags are set to 0.

DEF Flag

The DEF flag is set to 1 when an error has been detected during the DLC check. This flag can be cleared to 0 by writing 0 by the program.

16.3.2.9 RSCAN0GTINTSTS0 — Global TX Interrupt Status Register 0

Access: This register can be read in 8-, 16-, or 32-bit units.

Address: <RS-CAN0_base> + 0460_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	THIF3	CFTIF3	TQIF3	TAIF3	TSIF3	—	—	—	THIF2	CFTIF2	TQIF2	TAIF2	TSIF2
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R*1	R*1	R*1	R*1	R*1	R	R	R	R*1	R*1	R*1	R*1	R*1

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	THIF1	CFTIF1	TQIF1	TAIF1	TSIF1	—	—	—	THIF0	CFTIF0	TQIF0	TAIF0	TSIF0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R*1	R*1	R*1	R*1	R*1	R	R	R	R*1	R*1	R*1	R*1	R*1

Note 1. The bit is automatically cleared in global reset or channel reset mode.

Table 16.23 RSCAN0GTINTSTS0 Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 29	—	Reserved These bits are always read as 0. When writing, write 0.
28	THIF3	TX History List Interrupt Channel 3 0: Channel 3 TX history list interrupt flag is not set. 1: Channel 3 TX history list interrupt flag is set.
27	CFTIF3	COM FIFO TX Mode Interrupt Flag Channel 3 0: Channel 3 COM FIFO TX mode interrupt flag is not set. 1: Channel 3 COM FIFO TX mode interrupt flag is set.
26	TQIF3	TX Queue Interrupt Flag Channel 3 0: Channel 3 TX queue interrupt flag is not set. 1: Channel 3 TX queue interrupt flag is set.
25	TAIF3	TX Abort Interrupt Flag Channel 3 0: Channel 3 TX abort interrupt flag is not set. 1: Channel 3 TX abort interrupt flag is set.
24	TSIF3	Transmit Buffer Interrupt Status Flag Channel 3 0: Channel 3 Transmit successful completion interrupt flag is not set. 1: Channel 3 Transmit successful completion interrupt flag is set.
23 to 21	—	Reserved These bits are always read as 0. When writing, write 0.
20	THIF2	TX History List Interrupt Channel 2 0: Channel 2 TX history list interrupt flag is not set. 1: Channel 2 TX history list interrupt flag is set.
19	CFTIF2	COM FIFO TX Mode Interrupt Flag Channel 2 0: Channel 2 COM FIFO TX mode interrupt flag is not set. 1: Channel 2 COM FIFO TX mode interrupt flag is set.
18	TQIF2	TX Queue Interrupt Flag Channel 2 0: Channel 2 TX queue interrupt flag is not set. 1: Channel 2 TX queue interrupt flag is set.
17	TAIF2	TX Abort Interrupt Flag Channel 2 0: Channel 2 TX abort interrupt flag is not set. 1: Channel 2 TX abort interrupt flag is set.
16	TSIF2	TX Success Interrupt Flag Channel 2 0: Channel 2 TX successful completion interrupt flag is not set. 1: Channel 2 TX successful completion interrupt flag is set.

Table 16.23 RSCAN0GTINTSTS0 Register Contents (2/2)

Bit Position	Bit Name	Function
15 to 13	—	Reserved These bits are always read as 0. When writing, write 0.
12	THIF1	TX History List Interrupt Channel 1 0: Channel 1 TX history list interrupt flag is not set. 1: Channel 1 TX history list interrupt flag is set.
11	CFTIF1	COM FIFO TX Mode Interrupt Flag Channel 1 0: Channel 1 COM FIFO TX mode interrupt flag is not set. 1: Channel 1 COM FIFO TX mode interrupt flag is set.
10	TQIF1	TX Queue Interrupt Flag Channel 1 0: Channel 1 TX queue interrupt flag is not set. 1: Channel 1 TX queue interrupt flag is set.
9	TAIF1	Transmit Buffer Abort Interrupt Status Flag Channel 1 0: Channel 1 Transmit buffer abort interrupt flag is not set. 1: Channel 1 Transmit buffer abort interrupt flag is set.
8	TSIF1	TX Success Interrupt Flag Channel 1 0: Channel 1 TX successful completion interrupt flag is not set. 1: Channel 1 TX successful completion interrupt flag is set.
7 to 5	—	Reserved These bits are always read as 0. When writing, write 0.
4	THIF0	TX History List Interrupt Channel 0 0: Channel 0 TX history list interrupt flag is not set. 1: Channel 0 TX history list interrupt flag is set.
3	CFTIF0	COM FIFO TX Mode Interrupt Flag Channel 0 0: Channel 0 COM FIFO TX mode interrupt flag is not set. 1: Channel 0 COM FIFO TX mode interrupt flag is set.
2	TQIF0	TX Queue Interrupt Flag Channel 0 0: Channel 0 TX queue interrupt flag is not set. 1: Channel 0 TX queue interrupt flag is set.
1	TAIF0	Transmit Buffer Abort Interrupt Flag Channel 0 0: Channel 0 Transmit buffer abort interrupt flag is not set. 1: Channel 0 Transmit buffer abort interrupt flag is set.
0	TSIF0	TX Success Interrupt Flag Channel 0 0: Channel 0 TX successful completion interrupt flag is not set. 1: Channel 0 TX successful completion interrupt flag is set.

Note 1. These bits are automatically cleared in global reset mode or channel reset mode.

TSIFm Bit

When the RSCAN0TMIECy.TMIE bit is set to 1 (transmit buffer interrupt is enabled) and corresponding RSCAN0TMSTSp.TMTRF[1:0] flags are set to 10_B (transmission has been completed, without transmit abort request) or 11_B (transmission has been completed, with transmit abort request), the TSIFm bit becomes 1.

The TSIFm flag becomes 0 by setting 00_B to all the TMTRF[1:0] flags which satisfy the condition to lead TSIFm = 1. The TSIFm flag also becomes 0 by setting the TMIE bit to 0.

TAIFm Bit

When RSCAN0CmCTR.TAIE bit is set to 1 (transmit abort interrupt is enabled) and the RSCAN0TMSTSp.TMTRF[1:0] flags are 01_B (transmit abort has been completed), the TAIFm bit becomes 1.

The TAIFm flag becomes 0 by setting 00_B to all the TMTRF[1:0] flags of which transmit abort has been completed.

TQIFm Bit

When the RSCAN0TXQCCm.TXQIE bit is set to 1 (transmit queue interrupt is enabled) and the RSCAN0TXQSTSm.TXQIF bit is set to 1 (a transmit queue interrupt request is present), the TQIFm bit becomes 1.

When the RSCAN0TXQSTSm.TXQIF bit (transmit queue interrupt request) is set to 0, the TQIFm bit becomes 0. The TQIFm flag also becomes 0 by setting the TXQIE bit to 0.

CFTIFm Bit

When the RSCAN0CFCCk.CFTXIE bit is set to 1 (transmit/receive FIFO transmission interrupt is enabled) and the RSCAN0CFSTSk.CFTXIF bit is set to 1 (a transmit/receive FIFO transmission interrupt request is present), the CFTIFm bit becomes 1.

The CFTIFm bit becomes 0 by setting all the RSCAN0CFSTSk.CFTXIF bits to 0. The CFTIFm flag also becomes 0 by setting the CFTXIE bit to 0.

THIFm Bit

When the RSCAN0THLCCm.THLIE bit is set to 1 (transmit history interrupt is enabled) and the RSCAN0THLSTSm.THLIF bit is set to 1 (a transmit history interrupt request is present), the THIFm bit becomes 1.

The THIFm bit becomes 0 by setting the RSCAN0THLSTSm.THLIF bit to 0. The THIFm flag also becomes 0 by setting the THLIE bit to 0.

16.3.2.10 RSCAN0GTSC — Global Timestamp Counter Register

Access: This register can be read in 16- or 32-bit units.

Address: <RS-CAN0_base> + 0094_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TS[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 16.24 RSCAN0GTSC Register Contents

Bit Position	Bit Name	Function
31 to 16	—	Reserved These bits are always read as 0.
15 to 8	TS[15:0]	Timestamp value The timestamp counter value can be read. Counter value: 0000_H to $FFFF_H$

TS[15:0] Bits

When the TS[15:0] bits are read, the read value shows the timestamp counter (16-bit free-running counter) value at that time. The TS[15:0] value at the time when the SOF of the received message is detected is stored in the receive buffer or the FIFO buffer. The timestamp counter is initialized in global reset mode.

Start timing and stop timing of the timestamp counter depend on the count source.

- When the TSSS bit value in the RSCAN0GCFG register is 0 (pclk):
  The timestamp counter starts counting when the RS-CAN module has transitioned to global operating mode. This counter stops counting when the RS-CAN module has transitioned to global stop mode or global test mode.
- When the TSSS bit value in the RSCAN0GCFG register is 1 (CANm bit time clock):
  The timestamp counter starts counting when the corresponding channel has transitioned to channel communication mode. This counter stops counting when the corresponding channel has transitioned to channel reset mode or channel halt mode.

16.3.2.11 RSCAN0GAFLECTR — Receive Rule Entry Control Register

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: <RS-CAN0_base> + 0098_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	AFL DAE	—	—	—	AFLPN[4:0]				
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Table 16.25 RSCAN0GAFLECTR Register Contents

Bit Position	Bit Name	Function
31 to 9	—	Reserved These bits are always read as 0. When writing, write 0.
8	AFLDAE	Receive Rule Table Write Enable 0: Writing to the receive rule table is disabled. 1: Writing to the receive rule table is enabled.
7 to 5	—	Reserved These bits are always read as 0. The write value should always be 0.
4 to 0	AFLPN [4:0]	Receive Rule Table Page Number Configuration A page number can be selected from a range of page 0 (00000_B) to page 15 (01111_B).

AFLDAE Bit

Setting this bit to 0 disables data writing to the receive rule table. After data writing to the receive rule table is completed, set this bit to 0 to disable writing to the receive rule table. The receive rule table can be read regardless of the value of this bit.

Set the AFLDAE bit to 1 in global reset mode.

AFLPN[4:0] Bits

These bits are used to set the page number of the receive rule table. Sixteen receive rules can be set per page.

Set these bits to a value within a range of 00000_B to 01111_B.

16.3.2.12 RSCAN0GAFLCFG0 — Receive Rule Configuration Register 0

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: <RS-CAN0_base> + $009C_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RNC0[7:0]								RNC1[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RNC2[7:0]								RNC3[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 16.26 RSCAN0GAFLCFG0 Register Contents

Bit Position	Bit Name	Function
31 to 24	RNC0[7:0]	Number of Channel 0 Rules The number of rules for channel 0
23 to 16	RNC1[7:0]	Number of Channel 1 Rules The number of rules for channel 1
15 to 8	RNC2[7:0]	Number of Channel 2 Rules The number of rules for channel 2
7 to 0	RNC3[7:0]	Number of Channel 3 Rules The number of rules for channel 3

Modify the RSCAN0GAFLCFG0 register in global reset mode.

Up to "64 × (number of channels)" rules can be registered to the receive rule tables in the entire module. The number of receive rules per channel should meet the following conditions.

- The maximum number of rules per channel is 128.
- The total of the number of rules allocated to each channel is not larger than the number of rules permitted to the entire module.

RNC0[7:0] Bits

These bits are used to set the number of rules to be registered in the channel 0 receive rule table.

Set these bits to a value within a range of 00_H to 80_H.

RNC1[7:0] Bits

These bits are used to set the number of rules to be registered in the channel 1 receive rule table.

Set these bits to a value within a range of 00_H to 80_H.

RNC2[7:0] Bits

These bits are used to set the number of rules to be registered in the channel 2 receive rule table.

Set these bits to a value within a range of 00_H to 80_H.

RNC3[7:0] Bits

These bits are used to set the number of rules to be registered in the channel 3 receive rule table.

Set these bits to a value within a range of 00_H to 80_H.

16.3.2.13 RSCAN0GAFLIDj — Receive Rule ID Register (j = 0 to 15)

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: <RS-CAN0_base> + 0500_H + (j × 0010_H)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAFLIDE	GAFLRTR	GAFLLB	GAFLID[28:16]												
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAFLID[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 16.27 RSCAN0GAFLIDj Register Contents

Bit Position	Bit Name	Function
31	GAFLIDE	IDE Select 0: Standard ID 1: Extended ID
30	GAFLRTR	RTR Select 0: Data frame 1: Remote frame
29	GAFLLB	Receive Rule Target Message Select 0: When a message transmitted from another CAN node is received 1: When a message transmitted from own node is received
28 to 0	GAFLID[28:0]	ID Set Set the ID of the receive rule. For the standard ID, set the ID in bits b10 to b0 and set bits b28 to b11 to 0.

Modify the RSCAN0GAFLIDj register in global reset mode when the AFLDAE bit in the RSCAN0GAFLECTR register is set to 1 (writing to the receive rule table is enabled).

GAFLIDE Bit

This bit is used to select the ID format (standard ID or extended ID) of the receive rule. This bit is compared with the IDE bit in the received message during the acceptance filter processing.

GAFLRTR Bit

This bit is used to select the frame format (data frame or remote frame) of the receive rule. This bit is compared with the RTR bit in the received message during the acceptance filter processing.

GAFLLB Bit

Setting this bit to 0 causes data processing to be performed using the receive rule when a message transmitted from another CAN node is received.

Setting this bit is to 1 when the mirror function is used causes data processing to be performed using the receive rule when a message transmitted from the own CAN node is received.

GAFLID[28:0] Bits

These bits are used to set the ID field of the receive rule. The ID set by these bits is compared with the ID in the received message during the acceptance filter processing.

16.3.2.14 RSCAN0GAFLMj — Receive Rule Mask Register (j = 0 to 15)

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: <RS-CAN0_base> + 0504_H + (j × 0010_H)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAFLIDEM	GAFLRTRM	—	GAFLIDM[28:16]												
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAFLIDM[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 16.28 RSCAN0GAFLMj Register Contents

Bit Position	Bit Name	Function
31	GAFLIDEM	IDE Mask 0: The IDE bit is not compared. 1: The IDE bit is compared.
30	GAFLRTRM	RTR Mask 0: The RTR bit is not compared. 1: The RTR bit is compared.
29	—	Reserved This bit is always read as 0. When writing, write 0.
28 to 0	GAFLIDM [28:0]	ID Mask 0: The corresponding ID bit is not compared. 1: The corresponding ID bit is compared.

Modify the RSCAN0GAFLMj register in global reset mode when the AFLDAE bit in the RSCAN0GAFLECTR register is set to 1 (writing to the receive rule table is enabled).

GAFLIDEM Bit

When this bit is set to 1, filter processing is performed only for messages of the ID format specified by the GAFLIDE bit of the RSCAN0GAFLIDj register.

When this bit is set to 0, it is regarded that all received messages have matched the specified ID. To set the GAFLIDEM bit to 0, set the GAFLIDM[28:0] bits to all 0 at the same time.

GAFLRTRM Bit

This bit is used to mask the RTR bit of the receive rule.

GAFLIDM[28:0] Bits

These bits are used to mask the corresponding ID bits of the receive rule.

16.3.2.15 RSCAN0GAFLP0j — Receive Rule Pointer 0 Register (j = 0 to 15)

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: <RS-CAN0_base> + 0508_H + (j × 0010_H)

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GAFLDLC[3:0]				GAFLPTR[11:0]											
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAFL RMV	GAFLRMDP[6:0]						—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R

Table 16.29 RSCAN0GAFLP0j Register Contents

Bit Position	Bit Name	Function
31 to 28	GAFLDLC [3:0]	Receive Rule DLC Set b31 b30 b29 b28 0 0 0 0 : DLC check is disabled. 0 0 0 1 : 1 data byte 0 0 1 0 : 2 data bytes 0 0 1 1 : 3 data bytes 0 1 0 0 : 4 data bytes 0 1 0 1 : 5 data bytes 0 1 1 0 : 6 data bytes 0 1 1 1 : 7 data bytes 1 X X X : 8 data bytes
27 to 16	GAFLPTR [11:0]	Receive Rule Label Set Set the 12-bit label information.
15	GAFLRMV	Receive Buffer Enable 0: No receive buffer is used. 1: A receive buffer is used.
14 to 8	GAFLRMDP [6:0]	Receive Buffer Number Select Set the receive buffer number to store received messages.
7 to 0	—	Reserved These bits are always read as 0. When writing, write 0.

Modify the RSCAN0GAFLP0j register in global reset mode when the AFLDAE bit in the RSCAN0GAFLECTR register is set to 1 (writing to the receive rule table is enabled).

GAFLDLC[3:0] Bits

These bits are used to set the minimum data length necessary for receiving messages. If the data length of a message that is being filtered is equal to or larger than the value set by the GAFLDLC[3:0] bits, the message passes the DLC check. Setting these bits to 0000_B disables the DLC check function, allowing messages with any data length to pass the DLC check.

GAFLPTR[11:0] Bits

These bits are used to set a 12-bit label to be attached to messages that have passed through the filter. A label is attached when a message is stored in the receive buffer or the FIFO buffer.

GAFLRMV Bit

When this bit is set to 1, received messages that have passed through the filter are stored in the receive buffer selected by the GAFLRMDP[6:0] bits.

GAFLRMDP[6:0] Bits

These bits are used to select the number of a receive buffer that stores received messages having passed through the filter when the GAFLRMV bit is set to 1. Set these bits to a value smaller than the value set by the NRXMB[7:0] bits in the RSCAN0RMNB register.

16.3.2.16 RSCAN0GAFLP1j — Receive Rule Pointer 1 Register (j = 0 to 15)

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: <RS-CAN0_base> + $050C_H$ + (j × 0010_H)

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	GAFLFDP[19:16]			
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GAFLFDP[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 16.30 RSCAN0GAFLP1j Register Contents

Bit Position	Bit Name	Function
31 to 20	—	Reserved These bits are always read as 0. When writing, write 0.
19 to 0	GAFLFDP [19:0]	FIFO Buffer y Select (y = 0 to 19) y = 0 to 7 0: Transmit FIFO buffer y is not selected. 1: Transmit FIFO buffer y is selected. y = 8 to 19 0: Transmit/receive FIFO buffer y-8 is not selected. 1: Transmit/receive FIFO buffer y-8 is selected.

Modify the RSCAN0GAFLP1j register in global reset mode when the AFLDAE bit in the RSCAN0GAFLECTR register is set to 1 (receive rule table write is enabled).

GAFLFDP [19:0] Bits

These bits are used to specify FIFO buffers that store received messages having passed through the filter. Up to eight FIFO buffers are selectable. However, when the GAFLRMV bit in the RSCAN0GAFLP0j register is set to 1 (a receive buffer is used), up to seven FIFO buffers are selectable. Selectable buffers are receive FIFO buffers and the transmit/receive FIFO buffers for which the CFM[1:0] bits in the RSCAN0CFCCk register are set to 00_B (receive mode) or 10_B (gateway mode).

16.3.2.17 RSCAN0RMNB — Receive Buffer Number Register

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: <RS-CAN0_base> + $00A4_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	NRXMB[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 16.31 RSCAN0RMNB Register Contents

Bit Position	Bit Name	Function
31 to 8	—	Reserved These bits are always read as 0. When writing, write 0.
7 to 0	NRXMB [7:0]	Receive Buffer Number Configuration Set the number of receive buffers. Set a value of 0 to 64.

Modify the RSCAN0RMNB register in global reset mode.

NRXMB[7:0] Bits

These bits are used to set the total number of receive buffers of the entire RS-CAN module. The maximum value is “16 × (number of channels)”.

Setting these bits to all 0 makes receive buffers unavailable.

16.3.2.18 RSCAN0RMNDy — Receive Buffer New Data Register y (y = 0, 1)

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: <RS-CAN0_base> + $00A8_H$ + (y × 0004_H)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RMNSq (q = y × 32 + 31)	RMNSq (q = y × 32 + 30)	RMNSq (q = y × 32 + 29)	RMNSq (q = y × 32 + 28)	RMNSq (q = y × 32 + 27)	RMNSq (q = y × 32 + 26)	RMNSq (q = y × 32 + 25)	RMNSq (q = y × 32 + 24)	RMNSq (q = y × 32 + 23)	RMNSq (q = y × 32 + 22)	RMNSq (q = y × 32 + 21)	RMNSq (q = y × 32 + 20)	RMNSq (q = y × 32 + 19)	RMNSq (q = y × 32 + 18)	RMNSq (q = y × 32 + 17)	RMNSq (q = y × 32 + 16)
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RMNSq (q = y × 32 + 15)	RMNSq (q = y × 32 + 14)	RMNSq (q = y × 32 + 13)	RMNSq (q = y × 32 + 12)	RMNSq (q = y × 32 + 11)	RMNSq (q = y × 32 + 10)	RMNSq (q = y × 32 + 9)	RMNSq (q = y × 32 + 8)	RMNSq (q = y × 32 + 7)	RMNSq (q = y × 32 + 6)	RMNSq (q = y × 32 + 5)	RMNSq (q = y × 32 + 4)	RMNSq (q = y × 32 + 3)	RMNSq (q = y × 32 + 2)	RMNSq (q = y × 32 + 1)	RMNSq (q = y × 32 + 0)
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 16.32 RSCAN0RMNDy Register Contents

Bit Position	Bit Name	Function
31 to 16	RMNSq	Receive Buffer Reception Complete Flag q (q = y × 32 + 31 to y × 32 + 16) 0: Receive buffer q contains no new message. 1: Receive buffer q contains a new message.
15 to 0	RMNSq	Receive Buffer Reception Complete Flag q (q = y × 32 + 15 to y × 32 + 0) 0: Receive buffer q contains no new message. 1: Receive buffer q contains a new message.

Write 0 to the RSCAN0RMNDy register in global operating mode or global test mode.

RMNSq Flags (q = 0 to 63)

Each flag is set to 1 when a process starts to store a message in the corresponding receive buffer.

To clear these flags to 0, write 0 by the program. In this case, write 0 to bits to be cleared and write 1 to other bits using the store instruction. Note that when using a bit-manipulation instruction for writing to this register, the read-modify-write operation may inadvertently clear the flags. These bits cannot be set to 0 while a message is being stored. Time for storing a message is 10 pclk clock cycles.

These flags are cleared to 0 in global reset mode.

16.3.2.19 RSCAN0RMIDq — Receive Buffer ID Register (q = 0 to 63)

Access: This register can be read in 8-, 16-, or 32-bit units.

Address: <RS-CAN0_base> + 0600_H + (q × 0010_H)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RMIDE	RMRTR	—	RMID[28:16]												
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RMID[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 16.33 RSCAN0RMIDq Register Contents

Bit Position	Bit Name	Function
31	RMIDE	Receive Buffer IDE 0: Standard ID 1: Extended ID
30	RMRTR	Receive Buffer RTR 0: Data frame 1: Remote frame
29	—	Reserved This bit is always read as 0.
28 to 0	RMID [28:0]	Receive Buffer ID Data The standard ID or extended ID of received message can be read. Read bits b10 to b0 for standard ID. Bits b28 to b11 are read as 0.

RMIDE Bit

This bit indicates the ID format (standard ID or extended ID) of the message stored in the receive buffer.

RMRTR Bit

This bit indicates the frame format (data frame or remote frame) of the message stored in the receive buffer.

RMID[28:0] Bits

These bits indicate the ID of the message stored in the receive buffer.

16.3.2.20 RSCAN0RMPTRq — Receive Buffer Pointer Register (q = 0 to 63)

Access: This register can be read in 8-, 16-, or 32-bit units.

Address: <RS-CAN0_base> + 0604_H + (q × 0010_H)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RMDLC[3:0]				RMPTR[11:0]											
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RMTS[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 16.34 RSCAN0RMPTRq Register Contents

Bit Position	Bit Name	Function
31 to 28	RMDLC [3:0]	Receive Buffer DLC Data b31 b30 b29 b28 0 0 0 0: No data bytes 0 0 0 1: 1 data byte 0 0 1 0: 2 data bytes 0 0 1 1: 3 data bytes 0 1 0 0: 4 data bytes 0 1 0 1: 5 data bytes 0 1 1 0: 6 data bytes 0 1 1 1: 7 data bytes 1 X X X: 8 data bytes
27 to 16	RMPTR [11:0]	Receive Buffer Label Data Label information of the received message can be read.
15 to 0	RMTS [15:0]	Receive Buffer Timestamp Data Timestamp value of the received message can be read.

RMDLC[3:0] Bits

These bits indicate the data length of the message stored in the receive buffer.

RMPTR[11:0] Bits

These bits indicate the label information of the message stored in the receive buffer.

RMTS[15:0] Bits

These bits indicate the timestamp value of the message stored in the receive buffer.

16.3.2.21 RSCAN0RMDF0q — Receive Buffer Data Field 0 Register (q = 0 to 63)

Access: This register can be read in 8-, 16-, or 32-bit units.

Address: <RS-CAN0_base> + 0608_H + (q × 0010_H)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RMDB3[7:0]								RMDB2[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RMDB1[7:0]								RMDB0[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 16.35 RSCAN0RMDF0q Register Contents

Bit Position	Bit Name	Function
31 to 24	RMDB3 [7:0]	Receive Buffer Data Byte 3 Data in the message stored in the receive buffer can be read.
23 to 16	RMDB2 [7:0]	Receive Buffer Data Byte 2 Data in the message stored in the receive buffer can be read.
15 to 8	RMDB1 [7:0]	Receive Buffer Data Byte 1 Data in the message stored in the receive buffer can be read.
7 to 0	RMDB0 [7:0]	Receive Buffer Data Byte 0 Data in the message stored in the receive buffer can be read.

When the RMDLC[3:0] value in the RSCAN0RMPTRq register is smaller than 1000_B, data bytes for which no data is set are read as 00_H.

16.3.2.22 RSCAN0RMDF1q — Receive Buffer Data Field 1 Register (q = 0 to 63)

Access: This register can be read in 8-, 16-, or 32-bit units.

Address: <RS-CAN0_base> + $060C_H$ + (q × 0010_H)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RMDB7[7:0]								RMDB6[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RMDB5[7:0]								RMDB4[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 16.36 RSCAN0RMDF1q Register Contents

Bit Position	Bit Name	Function
31 to 24	RMDB7 [7:0]	Receive Buffer Data Byte 7 Data in the message stored in the receive buffer can be read.
23 to 16	RMDB6 [7:0]	Receive Buffer Data Byte 6 Data in the message stored in the receive buffer can be read.
15 to 8	RMDB5 [7:0]	Receive Buffer Data Byte 5 Data in the message stored in the receive buffer can be read.
7 to 0	RMDB4 [7:0]	Receive Buffer Data Byte 4 Data in the message stored in the receive buffer can be read.

When the RMDLC[3:0] value in the RSCAN0RMPTRq register is smaller than 1000_B, data bytes for which no data is set are read as 00_H.

16.3.2.23 RSCAN0RFCCx — Receive FIFO Buffer Configuration/Control Register (x = 0 to 7)

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: <RS-CAN0_base> + $00B8_H$ + (x × 0004_H)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFIGCV[2:0]			RFIM	—	RFDC[2:0]			—	—	—	—	—	—	RFIE	RFE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R	R	R	R	R	R/W	R/W

Table 16.37 RSCAN0RFCCx Register Contents

Bit Position	Bit Name	Function
31 to 16	—	Reserved These bits are always read as 0. When writing, write 0.
15 to 13	RFIGCV [2:0]	Receive FIFO Interrupt Request Timing Select b15 b14 b13 0 0 0: When FIFO is 1/8 full. 0 0 1: When FIFO is 2/8 full. 0 1 0: When FIFO is 3/8 full. 0 1 1: When FIFO is 4/8 full. 1 0 0: When FIFO is 5/8 full. 1 0 1: When FIFO is 6/8 full. 1 1 0: When FIFO is 7/8 full. 1 1 1: When FIFO is full.
12	RFIM	Receive FIFO Interrupt Source Select 0: An interrupt occurs when the condition set by the RFIGCV[2:0] bits is met. 1: An interrupt occurs each time a message has been received.
11	—	Reserved This bit is always read as 0. When writing, write 0.
10 to 8	RFDC [2:0]	Receive FIFO Buffer Depth Configuration b10 b9 b8 0 0 0: 0 messages 0 0 1: 4 messages 0 1 0: 8 messages 0 1 1: 16 messages 1 0 0: 32 messages 1 0 1: 48 messages 1 1 0: 64 messages 1 1 1: 128 messages
7 to 2	—	Reserved These bits are always read as 0. When writing, write 0.
1	RFIE	Receive FIFO Interrupt Enable 0: Receive FIFO interrupt is disabled. 1: Receive FIFO interrupt is enabled.
0	RFE	Receive FIFO Buffer Enable 0: No receive FIFO buffer is used. 1: Receive FIFO buffers are used.

RFIGCV[2:0] Bits

These bits are used to select the timing of generating a receive FIFO interrupt request when the RFIM bit is set to 0. An interrupt request is generated when the number of stored messages reaches the specified ratio (in eighths) of the storable messages set by the RFDC[2:0] bits.

When the RFDC[2:0] bits are set to 001_B (4 messages), set the RFIGCV[2:0] bits to 001_B, 011_B, 101_B, or 111_B. Modify these bits in global reset mode.

RFIM Bit

This bit is used to select a FIFO interrupt source. Modify this bit in global reset mode.

RFDC[2:0] Bits

These bits are used to select the number of messages that can be stored in a single receive FIFO buffer. If these bits are set to 000_B, do not use any receive FIFO buffer. Modify these bits in global reset mode.

RFIE Bit

Setting the RFIE bit to 1 enables receive FIFO interrupts. Modify this bit with the RFE bit set to 0 (no receive FIFO buffer is used).

RFE Bit

Setting the RFE bit to 1 makes receive FIFO buffers available. Clearing this bit to 0 sets the RFEMP flag in the RSCAN0RFSTSx register to 1 (the receive FIFO buffer contains no unread message (buffer empty)). Modify this bit in global operating mode or global test mode.

16.3.2.24 RSCAN0RFSTSx — Receive FIFO Buffer Status Register (x = 0 to 7)

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: <RS-CAN0_base> + $00D8_H$ + (x × 0004_H)

Value after reset: $0000\ 0001_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFMC[7:0]								—	—	—	—	RFIF	RFMLT	RFFLL	RFEMP
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W*1	R/W*1	R	R

Note 1. The only effective value for writing to this flag bit is 0, which clears the status. Any other writing to the bit results in retention of the status and does not change the value.

Table 16.38 RSCAN0RFSTSx Register Contents

Bit Position	Bit Name	Function
31 to 16	—	Reserved These bits are always read as 0. When writing, write 0.
15 to 8	RFMC[7:0]	Receive FIFO Unread Message Counter The number of unread messages stored in the receive FIFO buffer is displayed.
7 to 4	—	Reserved These bits are always read as 0. When writing, write 0.
3	RFIF	Receive FIFO Interrupt Request Flag 0: No receive FIFO interrupt request is present. 1: A receive FIFO interrupt request is present.
2	RFMLT	Receive FIFO Message Lost Flag 0: No receive FIFO message is lost. 1: A receive FIFO message is lost.
1	RFFLL	Receive FIFO Buffer Full Status Flag 0: The receive FIFO buffer is not full. 1: The receive FIFO buffer is full.
0	RFEMP	Receive FIFO Buffer Empty Status Flag 0: The receive FIFO buffer contains unread messages. 1: The receive FIFO buffer contains no unread message (buffer empty).

RFMC[7:0] Flags

These flags indicate the number of unread messages in the receive FIFO buffer. These flags become 00_H when the RFE bit in the RSCAN0RFCCx register is set to 0.

RFIF Flag

This flag is set to 1 when the conditions for receive FIFO interrupt request generation set by the RFIGCV[2:0] bits and the RFIM bit in the RSCAN0RFCCx register are satisfied. This flag is cleared to 0 in global reset mode or by writing 0 to this flag. Modify this bit in global operating mode or global test mode.

RFMLT Flag

This flag is set to 1 if it is attempted to store a new message when the receive FIFO buffer is full. In this case, the new message is discarded.

This flag is cleared to 0 in global reset mode or by writing 0 to this flag.

Modify this bit in global operating mode or global test mode.

RFFLL Flag

This flag is set to 1 when the number of messages stored in the receive FIFO buffer matches the FIFO buffer depth set by the RFDC[2:0] bits in the RSCAN0RFCCx register.

If the number of messages stored in the receive FIFO buffer becomes smaller than the FIFO buffer depth set by the RFDC[2:0] bits, this flag is cleared to 0. This flag is also cleared to 0 when the RFE bit in the RSCAN0RFCCx register is set to 0 (no receive FIFO buffer is used) or in global reset mode.

RFEMP Flag

This flag is set to 1 when all messages in the receive FIFO buffer have been read. This flag is also set to 1 when the RFE bit in the RSCAN0RFCCx register is 0 or in global reset mode.

This flag is cleared to 0 when even a single received message has been stored in the receive FIFO buffer.

NOTE

To clear the RFMLT or RFIF flag to 0, use a store instruction to write “0” to the given flag and “1” to the other flags. Note that when using a bit-manipulation instruction for writing to this register, the read-modify-write operation may inadvertently clear the flags.

16.3.2.25 RSCAN0RFPCTRx — Receive FIFO Buffer Pointer Control Register (x = 0 to 7)

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: <RS-CAN0_base> + $00F8_H$ + (x × 0004_H)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	RFPC[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	W	W	W	W	W	W	W	W

Table 16.39 RSCAN0RFPCTRx Register Contents

Bit Position	Bit Name	Function
31 to 8	—	Reserved When writing, write 0.
7 to 0	RFPC [7:0]	Receive FIFO Pointer Control When FF_H is written to these bits, the read pointer moves to the next unread message in the receive FIFO buffer.

RFPC[7:0] Bits

When FF_H is written to the RFPC[7:0] bits, the read pointer moves to the next unread message in the receive FIFO buffer. At this time, the RFMC[7:0] (receive FIFO unread message counter) value in the RSCAN0RFSTSx register is decremented by 1. Read the RSCAN0RFID, RSCAN0RFPTR, RSCAN0RFDF0, and RSCAN0RFDF1 registers to read messages in the receive FIFO buffer, and then write FF_H to the RFPC[7:0] bits.

Write FF_H to these bits when the RFE bit in the RSCAN0RFCCx register is set to 1 (receive FIFO buffers are used) and the RFEMP flag in the RSCAN0RFSTSx register is 0 (the receive FIFO buffer contains unread messages).

16.3.2.26 RSCAN0RFIDx — Receive FIFO Buffer Access ID Register (x = 0 to 7)

Access: This register can be read in 8-, 16-, or 32-bit units.

Address: <RS-CAN0_base> + $0E00_H$ + (x × 0010_H)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RFIDE	RFRTR	—	RFID[28:16]												
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFID[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 16.40 RSCAN0RFIDx Register Contents

Bit Position	Bit Name	Function
31	RFIDE	Receive FIFO Buffer IDE 0: Standard ID 1: Extended ID
30	RFRTR	Receive FIFO Buffer RTR 0: Data frame 1: Remote frame
29	—	Reserved This bit is always read as 0.
28 to 0	RFID [28:0]	Receive FIFO Buffer ID Data The standard ID or extended ID of received message can be read. Read bits b10 to b0 for standard ID. Bits b28 to b11 are read as 0.

RFIDE Bit

This bit indicates the ID format (standard ID or extended ID) of the message stored in the receive FIFO buffer.

RFRTR Bit

This bit indicates the frame format (data frame or remote frame) of the message stored in the receive FIFO buffer.

RFID[28:0] Bits

These bits indicate the ID of the message stored in the receive FIFO buffer.

16.3.2.27 RSCAN0RFPTRx — Receive FIFO Buffer Access Pointer Register (x = 0 to 7)

Access: This register can be read in 8-, 16-, or 32-bit units.

Address: <RS-CAN0_base> + $0E04_H$ + (x × 0010_H)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RFDLC[3:0]				RFPTR[11:0]											
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFTS[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 16.41 RSCAN0RFPTRx Register Contents

Bit Position	Bit Name	Function
31 to 28	RFDLC [3:0]	Receive FIFO Buffer DLC Data b31 b30 b29 b28 0 0 0 0 : 0 data bytes 0 0 0 1 : 1 data byte 0 0 1 0 : 2 data bytes 0 0 1 1 : 3 data bytes 0 1 0 0 : 4 data bytes 0 1 0 1 : 5 data bytes 0 1 1 0 : 6 data bytes 0 1 1 1 : 7 data bytes 1 X X X : 8 data bytes
27 to 16	RFPTR [11:0]	Receive FIFO Buffer Label Data Label information of the received message can be read.
15 to 0	RFTS [15:0]	Receive FIFO Buffer Timestamp Data Timestamp value of the received message can be read.

RFDLC[3:0] Bits

These bits indicate the data length of the message stored in the receive FIFO buffer.

RFPTR[11:0] Bits

These bits indicate the label information of the message stored in the receive FIFO buffer.

RFTS[15:0] Bits

These bits indicate the timestamp value of the message stored in the receive FIFO buffer.

16.3.2.28 RSCAN0RFDF0x — Receive FIFO Buffer Access Data Field 0 Register (x = 0 to 7)

Access: This register can be read in 8-, 16-, or 32-bit units.

Address: <RS-CAN0_base> + $0E08_H$ + (x × 0010_H)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RFDB3[7:0]								RFDB2[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFDB1[7:0]								RFDB0[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 16.42 RSCAN0RFDF0x Register Contents

Bit Position	Bit Name	Function
31 to 24	RFDB3 [7:0]	Receive FIFO Buffer Data Byte 3
23 to 16	RFDB2 [7:0]	Receive FIFO Buffer Data Byte 2
15 to 8	RFDB1 [7:0]	Receive FIFO Buffer Data Byte 1
7 to 0	RFDB0 [7:0]	Receive FIFO Buffer Data Byte 0 Data in the message stored in the receive FIFO buffer can be read.

When the RFDLC[3:0] value in the RSCAN0RFPTRx register is smaller than 1000_B, data bytes for which no data is set are read as 00_H.

16.3.2.29 RSCAN0RFDF1x — Receive FIFO Buffer Access Data Field 1 Register (x = 0 to 7)

Access: This register can be read in 8-, 16-, or 32-bit units.

Address: <RS-CAN0_base> + $0E0C_H$ + (x × 0010_H)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RFDB7[7:0]								RFDB6[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFDB5[7:0]								RFDB4[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 16.43 RSCAN0RFDF1x Register Contents

Bit Position	Bit Name	Function
31 to 24	RFDB7 [7:0]	Receive FIFO Buffer Data Byte 7
23 to 16	RFDB6 [7:0]	Receive FIFO Buffer Data Byte 6
15 to 8	RFDB5 [7:0]	Receive FIFO Buffer Data Byte 5
7 to 0	RFDB4 [7:0]	Receive FIFO Buffer Data Byte 4 Data in the message stored in the receive FIFO buffer can be read.

When the RFDLC[3:0] value in the RSCAN0RFPTRx register is smaller than 1000_B, data bytes for which no data is set are read as 00_H.

16.3.2.30 RSCAN0CFCCk — Transmit/Receive FIFO Buffer Configuration/Control Register (k = 0 to 11)

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: <RS-CAN0_base> + 0118_H + (k × 0004_H)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CFITT[7:0]								CFTML[3:0]				CFITR	CFITSS	CFM[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFIGCV[2:0]			CFIM	—	CFDC[2:0]			—	—	—	—	—	CFTXIE	CFRXIE	CFE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W

Table 16.44 RSCAN0CFCCk Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 24	CFITT[7:0]	Set a message transmission interval. Set value: 00_H to FF_H
23 to 20	CFTML[3:0]	Transmit Buffer Link Configuration Set the transmit buffer number to be linked to the transmit/receive FIFO buffer.
19	CFITR	Transmit/Receive FIFO Interval Timer Resolution 0: Clock obtained by dividing pclk/2 by the ITRCP[15:0] bits x1 1: Clock obtained by dividing pclk/2 by the ITRCP[15:0] bits x10
18	CFITSS	Transmit/Receive FIFO Interval Timer Source Select 0: Clock selected by the CFITR bit (x1 / x10 cycles) 1: Bit time clock of the corresponding channel (FIFO is linked to the fixed channel.)
17 to 16	CFM[1:0]	Transmit/Receive FIFO Mode Select b17 b16 0 0: Receive mode 0 1: Transmit mode 1 0: Gateway mode 1 1: Setting prohibited
15 to 13	CFIGCV [2:0]	Transmit/Receive FIFO Reception Interrupt Request Timing Select b15 b14 b13 0 0 0: When FIFO is 1/8 full. 0 0 1: When FIFO is 2/8 full. 0 1 0: When FIFO is 3/8 full. 0 1 1: When FIFO is 4/8 full. 1 0 0: When FIFO is 5/8 full. 1 0 1: When FIFO is 6/8 full. 1 1 0: When FIFO is 7/8 full. 1 1 1: When FIFO is full.

Table 16.44 RSCAN0CFCCk Register Contents (2/2)

Bit Position	Bit Name	Function
12	CFIM	Transmit/Receive FIFO Interrupt Source Select 0: • Receive mode / gateway mode When the number of received messages reaches the condition set by the CFIGCV[2:0] bits, a FIFO reception interrupt request is generated. • Transmit mode / gateway mode When the buffer becomes empty upon completion of message transmission, a FIFO transmission interrupt request is generated. 1: • Receive mode / gateway mode A FIFO reception interrupt request is generated each time a message has been received. • Transmit mode / gateway mode A FIFO transmission interrupt request is generated each time a message has been transmitted.
11	—	Reserved This bit is always read as 0. When writing, write 0.
10 to 8	CFDC [2:0]	Transmit/Receive FIFO Buffer Depth Configuration b10 b9 b8 0 0 0: 0 messages 0 0 1: 4 messages 0 1 0: 8 messages 0 1 1: 16 messages 1 0 0: 32 messages 1 0 1: 48 messages 1 1 0: 64 messages 1 1 1: 128 messages
7 to 3	—	Reserved These bits are always read as 0. When writing, write 0.
2	CFTXIE	Transmit/Receive FIFO Transmission Interrupt Enable 0: Transmit/receive FIFO transmission interrupt is disabled. 1: Transmit/receive FIFO transmission interrupt is enabled.
1	CFRXIE	Transmit/Receive FIFO Reception Interrupt Enable 0: Transmit/receive FIFO reception interrupt is disabled. 1: Transmit/receive FIFO reception interrupt is enabled.
0	CFE	Transmit/Receive FIFO Buffer Enable 0: No transmit/receive FIFO buffer is used. 1: Transmit/receive FIFO buffers are used.

CFITT[7:0] Bits

These bits are used to set a message transmission interval for continuous message transmission from a transmit/receive FIFO buffer whose CFM[1:0] bits are set to 01_B (transmit mode) or 10_B (gateway mode).

Clear the CFE bit to 0 (no transmit/receive FIFO buffer is used) before modifying the CFITT[7:0] bits.

CFTML[3:0] Bits

These bits are used to set the number of the transmission buffer to be linked to the transmission and reception FIFO buffers when the CFM[1:0] bits are set to 01B (transmission mode) or 10B (gateway mode). Three transmission and reception FIFO buffers are available per channel and channel number m to which FIFO buffer k is allocated is the integer part of the quotient k/3. The number p of the transmission buffer to be linked to FIFO buffer k will be (16 × m) + CFTML[3:0].

For the relationship between transmission and reception FIFO buffer k and transmission buffer p, see **Table 16.11** and **Table 16.12**.

Setting the CFDC[2:0] bits to 001_B or greater enables the setting of the CFTML[3:0] bits.

Do not make a link to the transmission queue of the same channel or any transmission buffer allocated to the transmission queue. Modify these bits in global reset mode.

CFITR Bit

This bit is enabled when the CFITSS bit is 0.

When this bit is 0, the interval timer clock source is selected as the pclk/2 divided by the value of the ITRCP [15:0] bits of the RSCAN0GCFG register.

When this bit is 1, the interval timer clock source is selected as the pclk/2 divided by (ITRCP [15:0] bits × 10) of the RSCAN0GCFG register.

Modify this bit with the CFE bit set to 0 (no transmit/receive FIFO buffer is used).

CFITSS Bit

When this bit is 0, the clock selected by the CFITR bit is the count source of the interval timer.
When this bit is 1, the bit time clock of the channel that is liked to FIFO is the count source of the interval timer.

Modify this bit with the CFE bit set to 0 (no transmit/receive FIFO buffer is used).

CFM[1:0] Bits

These bits are used to select a transmit/receive FIFO mode. Modify these bits in global reset mode.

CFIGCV[2:0] Bits

These bits are used to select the timing of generating a transmit/receive FIFO reception interrupt request when the CFM[1:0] bits are set to 00_B (receive mode) or 10_B (gateway mode) and the CFIM bit is set to 0. An interrupt request is generated when the number of stored messages reaches the specified ratio (in eighths) of the storable messages set by the CFDC[2:0] bits.

When the CFDC[2:0] bits are set to 001_B (4 messages), set the CFIGCV[2:0] bits to 001_B, 011_B, 101_B, or 111_B.

Modify this bit in global reset mode.

CFIM Bits

This bit is used to select a transmit/receive FIFO interrupt source. Modify this bit in global reset mode.

CFDC[2:0] Bit

These bits are used to set the number of messages that can be stored in a single transmit/receive FIFO buffer. If no transmit/receive FIFO buffer is used, set these bits to 000_B. Modify these bits in global reset mode.

CFTXIE Bit

When this bit is set to 1 and the CFTXIF flag in the CFSTSk register becomes 1, a transmit/receive FIFO transmission interrupt request is generated.

Modify this bit with the CFE bit set to 0 (no transmit/receive FIFO buffer is used).

CFRXIE Bit

When this bit is set to 1 and the CFTXIF flag in the RSCAN0CFSTSk register becomes 1, a transmit/receive FIFO reception interrupt request is generated.

Modify this bit with the CFE bit set to 0.

CFE Bit

Setting this bit to 1 makes transmit/receive FIFO buffers available.

When this bit is set to 0 in transmit mode or gateway mode, if a message in the transmit/receive FIFO buffer is being transmitted or to be transmitted next, the transmit/receive FIFO buffer becomes empty after completion of transmission, CAN bus error detection, or arbitration lost. In other cases or in receive mode, the transmit/receive FIFO buffer becomes empty immediately.

This bit is cleared to 0 when the following conditions are met.

- Receive mode: Global reset mode
- Transmit mode or gateway mode: Channel reset mode

Modify this bit in the following mode.

- Receive mode: Global operating mode or global test mode
- Transmit mode or gateway mode: Channel communication mode or channel halt mode

16.3.2.31 RSCAN0CFSTSk — Transmit/Receive FIFO Buffer Status Register (k = 0 to 11)

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: <RS-CAN0_base> + 0178_H + (k × 0004_H)

Value after reset: $0000\ 0001_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFMC[7:0]								—	—	—	CFTXIF	CFRXIF	CFMLT	CFFLL	CFEMP
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W*1	R/W*1	R/W*1	R	R

Note 1. The only effective value for writing to this flag bit is 0, which clears the status. Any other writing to the bit results in retention of the status and does not change the value.

Table 16.45 RSCAN0CFSTSk Register Contents

Bit Position	Bit Name	Function
31 to 16	—	Reserved These bits are always read as 0. When writing, write 0.
15 to 8	CFMC [7:0]	Transmit/Receive FIFO Message Counter The number of messages stored in the transmit/receive FIFO buffer is indicated.
7 to 5	—	Reserved These bits are always read as 0. When writing, write 0.
4	CFTXIF	Transmit/Receive FIFO Transmission Interrupt Request Flag 0: No transmit/receive FIFO transmission interrupt request is present. 1: A transmit/receive FIFO transmission interrupt request is present.
3	CFRXIF	Transmit/Receive FIFO Reception Interrupt Request Flag 0: No transmit/receive FIFO reception interrupt request is present. 1: A transmit/receive FIFO reception interrupt request is present.
2	CFMLT	Transmit/Receive FIFO Message Lost Flag 0: No transmit/receive FIFO message is lost. 1: A transmit/receive FIFO message is lost.
1	CFFLL	Transmit/Receive FIFO Buffer Full Status Flag 0: The transmit/receive FIFO buffer is not full. 1: The transmit/receive FIFO buffer is full.
0	CFEMP	Transmit/Receive FIFO Buffer Empty Status Flag 0: The transmit/receive FIFO buffer contains messages. 1: The transmit/receive FIFO buffer contains no message (buffer empty).

CFMC[7:0] Bits

The CFMC[7:0] bits indicate the following values that depend on the setting of the CFM[1:0] bits in the RSCAN0CFCCk register.

- When CFM[1:0] value is 01_B (transmit mode): Number of untransmitted messages in the buffer
- When CFM[1:0] value is 00_B (receive mode): Number of unread received messages in the buffer
- When CFM[1:0] value is 10_B (gateway mode): Number of untransmitted received messages in the buffer

These bits are cleared to 0 when any of the following conditions is met.

- When CFM[1:0] value is 00_B: global reset mode
- When CFM[1:0] value is 01_B or 10_B: channel reset mode

CFTXIF Flag

The CFTXIF flag is set to 1 when any of the following conditions is met.

- When the CFM[1:0] value is 01_B or 10_B and the source selected by the CFIM bit in the RSCAN0CFCCk register occurred.

The CFTXIF flag is set to 0 when any of the following conditions is met.

- When 0 is written in the CFTXIF flag.
- When the CFM[1:0] value is 00_B: global reset mode
- When the CFM[1:0] is 01_B or 10_B: channel reset mode

This flag is set to 0 in global operation mode or global reset mode.

CFRXIF Flag

The CFRXIF flag is set to 1 when any of the following conditions is met.

- When the CFM[1:0] value is 00_B or 10_B and the source selected by the CFIM bit in the RSCAN0CFCCk register occurred.

The CFRXIF flag is set to 0 when any of the following conditions is met.

- When 0 is written in the CFRXIF flag.
- When the CFM[1:0] value is 00_B: global reset mode
- When the CFM[1:0] is 01_B or 10_B: channel reset mode

This flag is set to 0 in global operation mode or global reset mode.

CFMLT Flag

The CFMLT flag is set to 1 when any of the following conditions is met.

- When transmit or receive FIFO buffer is full, and a new message is stored in this register. In this case, a new message is terminated.

The CFMLT flag is set to 0 when any of the following conditions is met.

- When 0 is written in the CFMLT flag.
- When the CFM[1:0] value is 00_B: global reset mode
- When the CFM[1:0] is 01_B or 10_B: channel reset mode

This flag is set to 0 in global operation mode or global reset mode.

CFFLL Flag

The CFFLL flag is set to 1 when any of the following conditions is met.

- When the number of messages stored in the transmit and receive FIFO buffer is equal to the FIFO buffer stages set by the CFDF[2:0] bits

The CFFLL flag is set to 0 when any of the following conditions is met.

- When the number of messages stored in the transmit and receive FIFO buffer is smaller than the FIFO buffer stages set by the CFDF[2:0] bits.
- When the CFE bit is 0 (no transmit/receive FIFO buffer is used): Not during transmit abort
- When the CFM[1:0] value is 00_B: global reset mode
- When the CFM[1:0] is 01_B or 10_B: channel reset mode

CFEMP Flag

The CFEMP flag is set to 1 when any of the following conditions is met.

- When the CFM[1:0] value is 00_B: All messages have been read or global reset mode.
- When the CFM[1:0] value is 01_B or 10_B: All messages have been transmitted or channel reset mode.
- When the CFE bit is 0 (no transmit/receive FIFO buffer is used): Not during transmit abort

The CFEMP flag is cleared to 0 when any of the following conditions is met.

- When the CFM[1:0] value is 00_B or 10_B: Any one of received messages has been stored in the transmit/receive FIFO buffer.
- When the CFM[1:0] value is 01_B: A value FF_H has been written to the RSCAN0CFPCTRk register after data was written to the RSCAN0CFIDk, RSCAN0CFPTRk, RSCAN0CFDF0k, and RSCAN0CFDF1k registers.

NOTE

To clear the CFTXIF, CFRXIF, and CFMLT flags to 0, the program must write 0 to the corresponding flag to be cleared. When writing 0, using a store instruction, set 1 to other flags. Note that when using a bit-manipulation instruction for writing to this register, the read-modify-write operation may inadvertently clear the flags.

16.3.2.32 RSCAN0CFPCTRk — Transmit/Receive FIFO Buffer Pointer Control Register (k = 0 to 11)

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: <RS-CAN0_base> + $01D8_H$ + (k × 0004_H)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	CFPC[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	W	W	W	W	W	W	W	W

Table 16.46 RSCAN0CFPCTRk Register Contents

Bit Position	Bit Name	Function
31 to 8	—	Reserved When writing, write 0.
7 to 0	CFPC [7:0]	Transmit/Receive FIFO Pointer Control • Receive mode: Writing FF_H to these bits moves the read pointer to the next unread message in the transmit/receive FIFO buffer. • Transmit mode: Writing FF_H to these bits moves the write pointer to the next stage of the transmit/receive FIFO buffer. • Gateway mode: Setting prohibited

CFPC[7:0] Bits

- Receive mode (CFM[1:0] value in the RSCAN0CFCCk register is 00_B):
  Writing FF_H to the CFPC[7:0] bits moves the read pointer to the next unread message in the transmit/receive FIFO buffer. At this time, the CFMC[7:0] value (transmit/receive FIFO message counter) in the RSCAN0CFSTSk register is decremented by 1. Read the RSCAN0CFIDk, RSCAN0CFPTRk, RSCAN0CFDF0k, and RSCAN0CFDF1k registers to read messages in the transmit/receive FIFO buffer, and then write FF_H to the CFPC[7:0] bits.
  Write FF_H to these bits when the CFE bit in the RSCAN0CFCCk register is set to 1 (transmit/receive FIFO buffers are used) and the CFEMP flag in the RSCAN0CFSTSk register is 0 (the transmit/receive FIFO buffer contains messages).
- Transmit mode (CFM[1:0] value in the RSCAN0CFCCk register is 01_B):
  Writing FF_H to the CFPC[7:0] bits stores the data written to the RSCAN0CFIDk, RSCAN0CFPTRk, RSCAN0CFDF0k, and RSCAN0CFDF1k registers in the transmit/receive FIFO buffer and moves the write pointer to the next stage of the transmit/receive FIFO buffer. At this time, the CFMC[7:0] value is incremented by 1. Write transmit messages to the RSCAN0CFIDk, RSCAN0CFPTRk, RSCAN0CFDF0k, and RSCAN0CFDF1k registers and then write FF_H to the CFPC[7:0] bits.
  Write FF_H to these bits when the CFE bit in the RSCAN0CFCCk register is set to 1 and the

CFFLL flag in the RSCAN0CFSTSk register is 0 (the transmit/receive FIFO buffer is not full).

- Gateway mode (CFM[1:0] value in the RSCAN0CFCCk register is 10_B):
  Setting prohibited

16.3.2.33 RSCAN0CFIDk — Transmit/Receive FIFO Buffer Access ID Register (k = 0 to 11)

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: <RS-CAN0_base> + $0E80_H$ + (k × 0010_H)

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CFIDE	CFRTR	THLEN	CFID[28:16]												
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFID[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 16.47 RSCAN0CFIDk Register Contents

Bit Position	Bit Name	Function
31	CFIDE	Transmit/Receive FIFO Buffer IDE bit 0: Standard ID 1: Extended ID
30	CFRTR	Transmit/Receive FIFO Buffer RTR bit 0: Data frame 1: Remote frame
29	THLEN	Transmit History Data Store Enable bit This bit is valid only when the CFM[1:0] value is 01_B (transmit mode). 0: Transmit history data is not stored in the buffer. 1: Transmit history data is stored in the buffer.
28 to 0	CFID [28:0]	Transmit/Receive FIFO Buffer ID Data • When CFM[1:0] value is 01_B (transmit mode): Set standard ID or extended ID. For standard ID, write an ID to bits b10 to b0 and write 0 to bits b28 to b11. • When CFM[1:0] value is 00_B (receive mode): Standard ID or extended ID in the received message can be read. For standard ID, read bits b10 to b0. Bits b28 to b11 are read as 0.

This register is writable only when the CFM[1:0] value of the RSCAN0CFCCk register is 01_B (transmit mode). This register is readable only when the CFM[1:0] value is 00_B (receive mode). This register should not be read or written when the CFM[1:0] value is 10_B (gateway mode).

CFIDE Bit

This bit indicates the ID format (standard ID or extended ID) of the received message stored in the transmit/receive FIFO buffer when the CFM[1:0] value is 00_B. When the CFM[1:0] value is 01_B, set the ID format of the message to be transmitted from the transmit/receive FIFO buffer.

CFRTR Bit

This bit indicates the data format (data frame or remote frame) of the received message stored in the transmit/receive FIFO buffer when the CFM[1:0] value is 00_B. When the CFM[1:0] value is 01_B, set the data format of the message to be transmitted from the transmit/receive FIFO buffer.

THLEN Bit

When this bit is set to 1, the transmit history data (label information, buffer number, and buffer type) of transmitted messages is stored in the transmit history buffer after transmission is completed.

This bit is enabled when the CFM[1:0] value is 01_B (transmit mode).

CFID[28:0] Bits

These bits indicate the ID of the received message stored in the transmit/receive FIFO buffer when the CFM[1:0] value is 00_B.

When the CFM[1:0] value is 01_B, set the ID of the message to be transmitted from the transmit/receive FIFO buffer.

16.3.2.34 RSCAN0CFPTRk — Transmit/Receive FIFO Buffer Access Pointer Register (k = 0 to 11)

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: <RS-CAN0_base> + $0E84_H$ + (k × 0010_H)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CFDLC[3:0]				CFPTR[11:0]											
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFTS[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 16.48 RSCAN0CFPTRk Register Contents

Bit Position	Bit Name	Function
31 to 28	CFDLC [3:0]	Transmit/Receive FIFO Buffer DLC Data b31 b30 b29 b28 0 0 0 0 : 0 data bytes 0 0 0 1 : 1 data byte 0 0 1 0 : 2 data bytes 0 0 1 1 : 3 data bytes 0 1 0 0 : 4 data bytes 0 1 0 1 : 5 data bytes 0 1 1 0 : 6 data bytes 0 1 1 1 : 7 data bytes 1 X X X : 8 data bytes
27 to 16	CFPTR [11:0]	Transmit/Receive FIFO Buffer Label Data • When CFM[1:0] value is 01_B (transmit mode): Set the label information to be stored in the transmit history buffer. Only CFPTR[7:0] bits are valid. • When CFM[1:0] value is 00_B (receive mode): The label information of the received message can be read.
15 to 0	CFTS [15:0]	Transmit/Receive FIFO Buffer Timestamp Data These bits are valid only when the CFM[1:0] value is 00_B (receive mode). The timestamp value of the received message can be read.

This register is writable only when the CFM[1:0] value of the RSCAN0CFCCk register is 01_B (transmit mode). This register is readable only when the CFM[1:0] value is 00_B (receive mode). This register should not be read or written when the CFM[1:0] value is 10_B (gateway mode).

CFDLC[3:0] Bits

These bits indicate the data length of the received message stored in the transmit/receive FIFO buffer when the CFM[1:0] value is 00_B. When the CFM[1:0] value is 01_B, set the data length of the message to be transmitted from the transmit/receive FIFO buffer. If 9-byte or more data length is set, 8-byte data is transmitted.

CFPTR[11:0] Bits

These bits indicate the label information attached to the received message stored in the transmit/receive FIFO buffer when the CFM[1:0] value is 00_B. When the CFM[1:0] value is 01_B, the CFPTR[7:0] value is stored in the transmit history buffer when message transmission has been completed.

CFTS[15:0] Bits

These bits indicate the timestamp value of the message stored in the transmit/receive FIFO buffer.

These bits are valid when the CFM[1:0] value is 00_B.

16.3.2.35 RSCAN0CFDF0k — Transmit/Receive FIFO Buffer Access Data Field 0 Register (k = 0 to 11)

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: <RS-CAN0_base> + $0E88_H$ + (k × 0010_H)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CFDB3[7:0]								CFDB2[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFDB1[7:0]								CFDB0[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 16.49 RSCAN0CFDF0k Register Contents

Bit Position	Bit Name	Function
31 to 24	CFDB3 [7:0]	Transmit/Receive FIFO Buffer Data Byte 3
23 to 16	CFDB2 [7:0]	Transmit/Receive FIFO Buffer Data Byte 2
15 to 8	CFDB1 [7:0]	Transmit/Receive FIFO Buffer Data Byte 1
7 to 0	CFDB0 [7:0]	Transmit/Receive FIFO Buffer Data Byte 0 • When CFM[1:0] value is 01_B (transmit mode): Set the data of the transmit/receive FIFO buffer. • When CFM[1:0] value is 00_B (receive mode): The message data stored in the transmit/receive FIFO buffer can be read.

This register is writable only when the CFM[1:0] value of the RSCAN0CFCCk register is 01_B (transmit mode). This register is readable only when the CFM[1:0] value is 00_B (receive mode). When the CFDLC[3:0] value in the RSCAN0CFPTRk register is smaller than 1000_B, data bytes for which no data is set are read as 00_H.

This register should not be read or written when the CFM[1:0] value is 10_B (gateway mode).

16.3.2.36 RSCAN0CFDF1k — Transmit/Receive FIFO Buffer Access Data Field 1 Register (k = 0 to 11)

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: <RS-CAN0_base> + $0E8C_H$ + (k × 0010_H)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CFDB7[7:0]								CFDB6[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CFDB5[7:0]								CFDB4[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 16.50 RSCAN0CFDF1k Register Contents

Bit Position	Bit Name	Function
31 to 24	CFDB7 [7:0]	Transmit/Receive FIFO Buffer Data Byte 7
23 to 16	CFDB6 [7:0]	Transmit/Receive FIFO Buffer Data Byte 6
15 to 8	CFDB5 [7:0]	Transmit/Receive FIFO Buffer Data Byte 5
7 to 0	CFDB4 [7:0]	Transmit/Receive FIFO Buffer Data Byte 4 • When CFM[1:0] value is 01_B (transmit mode): Set the data of the transmit/receive FIFO buffer. • When CFM[1:0] value is 00_B (receive mode): The message data stored in the transmit/receive FIFO buffer can be read.

This register is writable only when the CFM[1:0] value of the RSCAN0CFCCk register is 01_B (transmit mode). This register is readable only when the CFM[1:0] value is 00_B (receive mode). When the CFDLC[3:0] value in the RSCAN0CFPTRk register is smaller than 1000_B, data bytes for which no data is set are read as 00_H.

This register should not be read or written when the CFM[1:0] value is 10_B (gateway mode)

16.3.2.37 RSCAN0FESTS — FIFO Empty Status Register

Access: This register can be read in 8-, 16-, or 32-bit units.

Address: <RS-CAN0_base> + 0238_H

Value after reset: $007F\ FFFF_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	CF11 EMP	CF10 EMP	CF9 EMP	CF8 EMP
Value after reset	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CF7 EMP	CF6 EMP	CF5 EMP	CF4 EMP	CF3 EMP	CF2 EMP	CF1 EMP	CF0 EMP	RF7 EMP	RF6 EMP	RF5 EMP	RF4 EMP	RF3 EMP	RF2 EMP	RF1 EMP	RF0 EMP
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 16.51 RSCAN0FESTS Register Contents

Bit Position	Bit Name	Function
31 to 23	—	Reserved These bits are always read as 0.
22 to 20	—	Reserved These bits are always read as 1.
19	CF11EMP	Transmit/Receive FIFO Buffer Empty Status Flag 0: Transmit/receive FIFO buffer k contains a message. 1: Transmit/receive FIFO buffer k contains no message. (k = 0 to 11)
18	CF10EMP	
17	CF9EMP	
16	CF8EMP	
15	CF7EMP	
14	CF6EMP	
13	CF5EMP	
12	CF4EMP	
11	CF3EMP	
10	CF2EMP	
9	CF1EMP	
8	CF0EMP	
7	RF7EMP	Receive FIFO Buffer Empty Status Flag 0: Receive FIFO buffer x contains an unread message. 1: Receive FIFO buffer x contains no unread message. (x = 0 to 7)
6	RF6EMP	
5	RF5EMP	
4	RF4EMP	
3	RF3EMP	
2	RF2EMP	
1	RF1EMP	
0	RF0EMP	

The RSCAN0FESTS register is set to $007F\ FFFF_H$ in global reset mode.

CFkEMP Flag (k = 0 to 11)

The CFkEMP flag is set to 1 when the CFEMP flag in the RSCAN0CFSTSk register is set to 1 (the transmit/receive FIFO buffer contains no message). When the CFEMP flag is cleared to 0 (the transmit/receive FIFO buffer contains messages), the CFkEMP flag is cleared to 0.

RFxEMP Flag (x = 0 to 7)

The RFxEMP flag is set to 1 when the RFEMP flag in the RSCAN0RFSTSx register is set to 1 (the receive FIFO buffer contains no unread message). When the RFEMP flag is cleared to 0 (the receive FIFO buffer contains unread messages), the RFxEMP flag is cleared to 0.

16.3.2.38 RSCAN0FFSTS — FIFO Full Status Register

Access: This register can be read in 8-, 16-, or 32-bit units.

Address: <RS-CAN0_base> + $023C_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	CF11FLL	CF10FLL	CF9FLL	CF8FLL
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CF7FLL	CF6FLL	CF5FLL	CF4FLL	CF3FLL	CF2FLL	CF1FLL	CF0FLL	RF7FLL	RF6FLL	RF5FLL	RF4FLL	RF3FLL	RF2FLL	RF1FLL	RF0FLL
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 16.52 RSCAN0FFSTS Register Contents

Bit Position	Bit Name	Function
31 to 20	—	Reserved These bits are always read as 0.
19	CF11FLL	Transmit/Receive FIFO Buffer Full Status Flag 0: Transmit/receive FIFO buffer k is not full. 1: Transmit/receive FIFO buffer k is full. (k = 0 to 11)
18	CF10FLL	
17	CF9FLL	
16	CF8FLL	
15	CF7FLL	
14	CF6FLL	
13	CF5FLL	
12	CF4FLL	
11	CF3FLL	
10	CF2FLL	
9	CF1FLL	
8	CF0FLL	
7	RF7FLL	Receive FIFO Buffer Full Status Flag 0: Receive FIFO buffer x is not full. 1: Receive FIFO buffer x is full. (x = 0 to 7)
6	RF6FLL	
5	RF5FLL	
4	RF4FLL	
3	RF3FLL	
2	RF2FLL	
1	RF1FLL	
0	RF0FLL	

The RSCAN0FFSTS register is cleared to $0000\ 0000_H$ in global reset mode.

CFkFLL Flag (k = 0 to 11)

The CFkFLL flag is set to 1 when the CFFLL flag in the RSCAN0CFSTSk register is set to 1 (the transmit/receive FIFO buffer is full). When the CFFLL flag is cleared to 0, the CFkFLL flag is cleared to 0 (the transmit/receive FIFO buffer is not full).

RFxFLL Flag (x = 0 to 7)

The RFxFLL flag is set to 1 when the RFFLL flag in the RSCAN0RFSTSx register is set to 1 (the receive FIFO buffer is full). When the RFFLL flag is cleared to 0 (the receive FIFO buffer is not full), the RFxFLL flag is cleared to 0.

16.3.2.39 RSCAN0FMSTS — FIFO Message Lost Status Register

Access: This register can be read in 8-, 16-, or 32-bit units.

Address: <RS-CAN0_base> + 0240_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	CF11 MLT	CF10 MLT	CF9 MLT	CF8 MLT
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CF7 MLT	CF6 MLT	CF5 MLT	CF4 MLT	CF3 MLT	CF2 MLT	CF1 MLT	CF0 MLT	RF7 MLT	RF6 MLT	RF5 MLT	RF4 MLT	RF3 MLT	RF2 MLT	RF1 MLT	RF0 MLT
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 16.53 RSCAN0FMSTS Register Contents

Bit Position	Bit Name	Function
31 to 20	—	Reserved These bits are always read as 0.
19	CF11MLT	Transmit/Receive FIFO Buffer Message Lost Status Flag 0: No transmit/receive FIFO buffer k message is lost. 1: A transmit/receive FIFO buffer k message is lost. (k = 0 to 11)
18	CF10MLT	
17	CF9MLT	
16	CF8MLT	
15	CF7MLT	
14	CF6MLT	
13	CF5MLT	
12	CF4MLT	
11	CF3MLT	
10	CF2MLT	
9	CF1MLT	
8	CF0MLT	
7	RF7MLT	Receive FIFO Buffer Message Lost Status Flag 0: No receive FIFO buffer x message is lost. 1: A receive FIFO buffer x message is lost. (x = 0 to 7)
6	RF6MLT	
5	RF5MLT	
4	RF4MLT	
3	RF3MLT	
2	RF2MLT	
1	RF1MLT	
0	RF0MLT	

The RSCAN0FMSTS register is cleared to $0000\ 0000_H$ in global reset mode.

CFkMLT Flag (k = 0 to 11)

The CFkMLT flag is set to 1 when the CFMLT flag in the RSCAN0CFSTSk register is set to 1 (a transmit/receive FIFO message is lost). When the CFMLT flag is cleared to 0, the CFkMLT flag is cleared to 0.

RFxMLT Flag (x = 0 to 7)

The RFxMLT flag is set to 1 when the RFMLT flag in the RSCAN0RFSTSx register is set to 1 (a receive FIFO message is lost). When the RFMLT flag is cleared to 0, the RFxMLT flag is cleared to 0.

16.3.2.40 RSCAN0RFISTS — Receive FIFO Buffer Interrupt Flag Status Register

Access: This register can be read in 8-, 16-, or 32-bit units.

Address: <RS-CAN0_base> + 0244_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	RF7IF	RF6IF	RF5IF	RF4IF	RF3IF	RF2IF	RF1F	RF0F
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 16.54 RSCAN0RFISTS Register Contents

Bit Position	Bit Name	Function
31 to 8	—	Reserved These bits are always read as 0.
7	RF7IF	Receive FIFO Buffer Interrupt Request Status Flag 0: No receive FIFO buffer x interrupt request is present. 1: A receive FIFO buffer x interrupt request is present. (x = 0 to 7)
6	RF6IF	
5	RF5IF	
4	RF4IF	
3	RF3IF	
2	RF2IF	
1	RF1F	
0	RF0F	

The RSCAN0RFISTS register is cleared to $0000\ 0000_H$ in global reset mode.

RFxIF Flag (x = 0 to 7)

The RFxIF flag is set to 1 when the RFIF flag in the RSCAN0RFSTSx register is set to 1 (a receive FIFO interrupt request is present). When the RFIF flag is cleared to 0, the RFxIF flag is cleared to 0.

16.3.2.41 RSCAN0CFRISTS — Transmit/Receive FIFO Buffer Reception Interrupt Flag Status Register

Access: This register can be read in 8-, 16-, or 32-bit units.

Address: <RS-CAN0_base> + 0248_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	CF11 RXIF	CF10 RXIF	CF9 RXIF	CF8 RXIF	CF7 RXIF	CF6 RXIF	CF5 RXIF	CF4 RXIF	CF3 RXIF	CF2 RXIF	CF1 RXIF	CF0 RXIF
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 16.55 RSCAN0CFRISTS Register Contents

Bit Position	Bit Name	Function
31 to 12	—	Reserved These bits are always read as 0.
11	CF11RXIF	Transmit/Receive FIFO Buffer Reception Interrupt Request Status Flag 0: No transmit/receive FIFO buffer k reception interrupt request is present. 1: A transmit/receive FIFO buffer k reception interrupt request is present. (k = 0 to 11)
10	CF10RXIF	
9	CF9RXIF	
8	CF8RXIF	
7	CF7RXIF	
6	CF6RXIF	
5	CF5RXIF	
4	CF4RXIF	
3	CF3RXIF	
2	CF2RXIF	
1	CF1RXIF	
0	CF0RXIF	

The RSCAN0CFRISTS register is cleared to $0000\ 0000_H$ in global reset mode.

CFxRXIF Flag (x = 0 to 11)

The CFxRXIF flag is set to 1 when the CFRXIF flag in the RSCAN0CFSTSk register is set to 1 (a transmit/receive FIFO reception interrupt request is present). When the CFRXIF flag is cleared to 0, the CFxRXIF flag is cleared to 0.

16.3.2.42 RSCAN0CFTISTS — Transmit/Receive FIFO Buffer Transmission Interrupt Flag Status Register

Access: This register can be read in 8-, 16-, or 32-bit units.

Address: <RS-CAN0_base> + $024C_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	CF11 TXIF	CF10 TXIF	CF9 TXIF	CF8 TXIF	CF7 TXIF	CF6 TXIF	CF5 TXIF	CF4 TXIF	CF3 TXIF	CF2 TXIF	CH1 TXIF1	CF0 TXIF
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 16.56 RSCAN0CFTISTS Register Contents

Bit Position	Bit Name	Function
31 to 12	—	Reserved These bits are always read as 0.
11	CF11TXIF	Transmit/Receive FIFO Buffer Transmission Interrupt Request Status Flag 0: No transmit/receive FIFO buffer k transmission interrupt request is present. 1: A transmit/receive FIFO buffer k transmission interrupt request is present. (k = 0 to 11)
10	CF10TXIF	
9	CF9TXIF	
8	CF8TXIF	
7	CF7TXIF	
6	CF6TXIF	
5	CF5TXIF	
4	CF4TXIF	
3	CF3TXIF	
2	CF2TXIF	
1	CF1TXIF	
0	CF0TXIF	

The RSCAN0CFTISTS register is cleared to $0000\ 0000_H$ in global reset mode.

CFkTXIF Flag (k = 0 to 11)

The CFkTXIF flag is set to 1 when the CFTXIF flag in the RSCAN0CFSTSk register is set to 1 (a transmit/receive FIFO transmission interrupt request is present). When the CFTXIF flag is cleared to 0, the CFkTXIF flag is cleared to 0.

16.3.2.43 RSCAN0TMCp — Transmit Buffer Control Register (p = 0 to 63)

Access: This register can be read/written in 8 bits/unit

Address: <RS-CAN0_base> + 0250_H + (p × 0001_H)

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	TMOM	TMTAR	TMTR
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W*1	R/W*1

Note 1. The only effective value for writing to this bit is 1, which sets the bit. Any other writing to the bit results in retention of the status and does not change the value.

Table 16.57 RSCAN0TMCp Register Contents

Bit Position	Bit Name	Function
7 to 3	—	Reserved These bits are always read as 0. When writing, write 0.
2	TMOM	One-Shot Transmission Enable 0: One-shot transmission is disabled. 1: One-shot transmission is enabled.
1	TMTAR	Transmit Abort Request 0: Transmit abort is not requested. 1: Transmit abort is requested.
0	TMTR	Transmit Request 0: Transmission is not requested. 1: Transmission is requested.

When the RSCAN0TMCp register meets any of the following conditions, set it to 00_H.

- The RSCAN0TMCp register (p = m × 16 + of the CFTML[3:0] value) corresponds to the transmit buffer number selected by the CFTML[3:0] bits in the RSCAN0CFCCk register.
- The RSCAN0TMCp register (p = (m × 16 + 15) to (m × 16 + 15 the TXQDC[3:0] value)) corresponds to the transmit buffer allocated to the transmit queue by the TXQDC[3:0] bits in the RSCAN0TXQCCm register (m = 0 to 3).

Bits in the RSCAN0TMCp register are cleared to all 0 in channel reset mode. Modify the RSCAN0TMCp register in channel communication mode or channel halt mode.

TMOM Bit

Setting this bit to 1 enables one-shot transmission. If transmission fails, retransmission defined in the CAN protocol is not performed.

Modify the TMOM bit when the TMTRM flag in the RSCAN0TMSTSp register is set to 0. To set the TMOM bit to 1, also set the TMTR bit together.

TMTAR Bit

Setting this bit to 1 generates a transmit abort request for the message stored in the transmit buffer. However, a message that is being transmitted or to be transmitted next cannot be aborted.

When the TMTR bit is set to 1, the TMTAR bit can be set to 1.

The TMTAR bit is cleared to 0 when any of the following conditions is set. It is not cleared to 0 by writing 0 by the program.

- Transmission has been completed.
- Transmit abort has been completed.
- An error or arbitration lost has been detected.

If this bit becomes 0 at the timing when the program writes 1 to this bit, this bit becomes 0.

TMTR Bit

Setting this bit to 1 transmits the message stored in the transmit buffer.

The TMTR bit is cleared to 0 when any of the following conditions is met. It is not cleared to 0 by writing 0 by the program.

- Transmission has been completed.
- Transmit abort has been completed by setting the TMTAR bit to 1.
- An error or arbitration lost has been detected with the TMOM bit set to 1.

Set the TMTR bit to 1 when the TMTRF[1:0] flag in the RSCAN0TMSTSp register is 00_B.

16.3.2.44 RSCAN0TMSTSp — Transmit Buffer Status Register (p = 0 to 63)

Access: This register can be read/written in 8-bit units.

Address: <RS-CAN0_base> + $02D0_H$ + (p × 0001_H)

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	TMTARM	TMTRM	TMTRF[1:0]		TMTSTS
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R

Table 16.58 RSCAN0TMSTSp Register Contents

Bit Position	Bit Name	Function
7 to 5	—	Reserved These bits are always read as 0. When writing, write 0.
4	TMTARM	Transmit Buffer Transmit Abort Request Status Flag 0: No transmit abort request is present. 1: A transmit abort request is present.
3	TMTRM	Transmit Buffer Transmit Request Status Flag 0: No transmit request is present. 1: A transmit request is present.
2,1	TMTRF[1:0]	Transmit Buffer Transmit Result Flag b2 b1 0 0: Transmission is in progress or no transmit request is present. 0 1: Transmit abort has been completed. 1 0: Transmission has been completed (without transmit abort request). 1 1: Transmission has been completed (with transmit abort request).
0	TMTSTS	Transmit Buffer Transmit Status Flag 0: Transmission is not in progress. 1: Transmission is in progress.

The RSCAN0TMSTSp register is cleared to all 0 in channel reset mode.

TMTARM Flag

The TMTARM flag is set to 1 when the TMTAR bit in the RSCAN0TMCp register is set to 1, and is cleared to 0 when the TMTAR bit is set to 0.

TMTRM Flag

The TMTRM flag is set to 1 when the TMTR bit in the RSCAN0TMCp register is set to 1, and is cleared to 0 when the TMTR bit is set to 0.

TMTRF[1:0] Flag

This flag indicates the result of transmission from the transmit buffer.

00_B: Transmission is in progress or no transmit request is present.

01_B: Transmission from the transmit buffer was aborted.

10_B: Transmission has been completed with the TMTAR bit in the RSCAN0TMCp register set to 0 (transmit abort is not requested).

11_B: Transmission has been completed with the TMTAR bit in the RSCAN0TMCp register set to 1 (transmit abort is requested).

Write 00_B to the TMTRF[1:0] flag in channel communication mode or channel halt mode. Do not write any value other than 00_B to this flag.

TMTSTS Flag

This flag is set to 1 when transmission from the transmit buffer starts, and is cleared to 0 when transmission from the transmit buffer has been completed or stopped due to a bus error or arbitration lost.

16.3.2.45 RSCAN0TMTRSTSy — Transmit Buffer Transmit Request Status Register y (y = 0, 1)

Access: This register can be read in 8-, 16-, or 32-bit units.

Address: <RS-CAN0_base> + 0350_H + (y × 0004_H)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMTRSTSp (p = y × 32 + 31)	TMTRSTSp (p = y × 32 + 30)	TMTRSTSp (p = y × 32 + 29)	TMTRSTSp (p = y × 32 + 28)	TMTRSTSp (p = y × 32 + 27)	TMTRSTSp (p = y × 32 + 26)	TMTRSTSp (p = y × 32 + 25)	TMTRSTSp (p = y × 32 + 24)	TMTRSTSp (p = y × 32 + 23)	TMTRSTSp (p = y × 32 + 22)	TMTRSTSp (p = y × 32 + 21)	TMTRSTSp (p = y × 32 + 20)	TMTRSTSp (p = y × 32 + 19)	TMTRSTSp (p = y × 32 + 18)	TMTRSTSp (p = y × 32 + 17)	TMTRSTSp (p = y × 32 + 16)
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMTRSTSp (p = y × 32 + 15)	TMTRSTSp (p = y × 32 + 14)	TMTRSTSp (p = y × 32 + 13)	TMTRSTSp (p = y × 32 + 12)	TMTRSTSp (p = y × 32 + 11)	TMTRSTSp (p = y × 32 + 10)	TMTRSTSp (p = y × 32 + 9)	TMTRSTSp (p = y × 32 + 8)	TMTRSTSp (p = y × 32 + 7)	TMTRSTSp (p = y × 32 + 6)	TMTRSTSp (p = y × 32 + 5)	TMTRSTSp (p = y × 32 + 4)	TMTRSTSp (p = y × 32 + 3)	TMTRSTSp (p = y × 32 + 2)	TMTRSTSp (p = y × 32 + 1)	TMTRSTSp (p = y × 32 + 0)
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 16.59 RSCAN0TMTRSTSy Register Contents

Bit Position	Bit Name	Function
31 to 16	TMTRSTSp	Transmit Buffer Transmit Request Status Flag p (p = y × 32 + 31 to y × 32 + 16) 0: No transmit request is present. 1: A transmit request is present.
15 to 0	TMTRSTSp	Transmit Buffer Transmit Request Status Flag p (p = y × 32 + 15 to y × 32 + 0) 0: No transmit request is present. 1: A transmit request is present.

TMTRSTSp Flags (p = 0 to 63)

These flags indicate the status of the TMTR bit in the RSCAN0TMCp register.

When the TMTR bit is set to 1 (transmission is requested), the corresponding TMTRSTSp flag is set to 1.

The corresponding TMTRSTSp flag is cleared to 0 when the TMTR bit is set to 0 (transmission is not requested) or in channel reset mode.

Table 16.60 shows the bit allocation.

Table 16.60 TMTRSTSp Bit Allocation (1/2)

Bit	Channel	Transmit Buffer No.
0	0	0
1	0	1
. .	. .	. .
15	0	15
16	1	0
. .	. .	. .
30	1	14
31	1	15
32	2	0
33	2	1

Table 16.60 TMTRSTSp Bit Allocation (2/2)

Bit	Channel	Transmit Buffer No.
. .	. .	. .
47	2	15
48	3	0
. .	. .	. .
62	3	14
63	3	15

16.3.2.46 RSCAN0TMTARSTSy — Transmit Buffer Transmit Abort Request Status Register (y = 0, 1)

Access: This register can be read in 8-, 16-, or 32-bit units.

Address: <RS-CAN0_base> + 0360_H + (y × 0004_H)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMTARSTSp (p = y × 32 + 31)	TMTARSTSp (p = y × 32 + 30)	TMTARSTSp (p = y × 32 + 29)	TMTARSTSp (p = y × 32 + 28)	TMTARSTSp (p = y × 32 + 27)	TMTARSTSp (p = y × 32 + 26)	TMTARSTSp (p = y × 32 + 25)	TMTARSTSp (p = y × 32 + 24)	TMTARSTSp (p = y × 32 + 23)	TMTARSTSp (p = y × 32 + 22)	TMTARSTSp (p = y × 32 + 21)	TMTARSTSp (p = y × 32 + 20)	TMTARSTSp (p = y × 32 + 19)	TMTARSTSp (p = y × 32 + 18)	TMTARSTSp (p = y × 32 + 17)	TMTARSTSp (p = y × 32 + 16)
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMTARSTSp (p = y × 32 + 15)	TMTARSTSp (p = y × 32 + 14)	TMTARSTSp (p = y × 32 + 13)	TMTARSTSp (p = y × 32 + 12)	TMTARSTSp (p = y × 32 + 11)	TMTARSTSp (p = y × 32 + 10)	TMTARSTSp (p = y × 32 + 9)	TMTARSTSp (p = y × 32 + 8)	TMTARSTSp (p = y × 32 + 7)	TMTARSTSp (p = y × 32 + 6)	TMTARSTSp (p = y × 32 + 5)	TMTARSTSp (p = y × 32 + 4)	TMTARSTSp (p = y × 32 + 3)	TMTARSTSp (p = y × 32 + 2)	TMTARSTSp (p = y × 32 + 1)	TMTARSTSp (p = y × 32 + 0)
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 16.61 RSCAN0TMTARSTSy Register Contents

Bit Position	Bit Name	Function
31 to 16	TMTARSTSp	Transmit Buffer Transmit Abort Request Status Flag p (p = y × 32 + 31 to y × 32 + 16) 0: No transmit abort request is present. 1: A transmit abort request is present.
15 to 0	TMTARSTSp	Transmit Buffer Transmit Abort Request Status Flag p (p = y × 32 + 15 to y × 32 + 0) 0: No transmit abort request is present. 1: A transmit abort request is present.

TMTARSTSp Flags (p = 0 to 63)

These flags indicate the status of the TMTAR bit in the RSCAN0TMCp register.

When the TMTAR bit is set to 1 (transmit abort is requested), the corresponding TMTARSTSp flag is set to 1.

The corresponding TMTARSTSp flag is cleared to 0 when the TMTAR bit is set to 0 (transmit abort is not requested) or in channel reset mode.

Table 16.62 shows the bit allocation.

Table 16.62 TMTARSTSp Bit Allocation (1/2)

Bit Position	Channel	Transmit Buffer No.
0	0	0
1	0	1
. .	. .	. .
15	0	15
16	1	0
. .	. .	. .
30	1	14
31	1	15
32	2	0

Table 16.62 TMTARSTSp Bit Allocation (2/2)

Bit Position	Channel	Transmit Buffer No.
33	2	1
. .	. .	. .
47	2	15
48	3	0
. .	. .	. .
62	3	14
63	3	15

16.3.2.47 RSCAN0TMTCSTSy — Transmit Buffer Transmit Complete Status Register (y = 0, 1)

Access: This register can be read in 8-, 16-, or 32-bit units.

Address: <RS-CAN0_base> + 0370_H + (y × 0004_H)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMTCSTSp (p = y × 32 + 31)	TMTCSTSp (p = y × 32 + 30)	TMTCSTSp (p = y × 32 + 29)	TMTCSTSp (p = y × 32 + 28)	TMTCSTSp (p = y × 32 + 27)	TMTCSTSp (p = y × 32 + 26)	TMTCSTSp (p = y × 32 + 25)	TMTCSTSp (p = y × 32 + 24)	TMTCSTSp (p = y × 32 + 23)	TMTCSTSp (p = y × 32 + 22)	TMTCSTSp (p = y × 32 + 21)	TMTCSTSp (p = y × 32 + 20)	TMTCSTSp (p = y × 32 + 19)	TMTCSTSp (p = y × 32 + 18)	TMTCSTSp (p = y × 32 + 17)	TMTCSTSp (p = y × 32 + 16)
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMTCSTSp (p = y × 32 + 15)	TMTCSTSp (p = y × 32 + 14)	TMTCSTSp (p = y × 32 + 13)	TMTCSTSp (p = y × 32 + 12)	TMTCSTSp (p = y × 32 + 11)	TMTCSTSp (p = y × 32 + 10)	TMTCSTSp (p = y × 32 + 9)	TMTCSTSp (p = y × 32 + 8)	TMTCSTSp (p = y × 32 + 7)	TMTCSTSp (p = y × 32 + 6)	TMTCSTSp (p = y × 32 + 5)	TMTCSTSp (p = y × 32 + 4)	TMTCSTSp (p = y × 32 + 3)	TMTCSTSp (p = y × 32 + 2)	TMTCSTSp (p = y × 32 + 1)	TMTCSTSp (p = y × 32 + 0)
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 16.63 RSCAN0TMTCSTSy Register Contents

Bit Position	Bit Name	Function
31 to 16	TMTCSTSp	Transmit Buffer Transmit Complete Status Flag p (p = y × 32 + 31 to y × 32 + 16) 0: Transmission has not been completed. 1: Transmission has been completed.
15 to 0	TMTCSTSp	Transmit Buffer Transmit Complete Status Flag p (p = y × 32 + 15 to y × 32 + 0) 0: Transmission has not been completed. 1: Transmission has been completed.

TMTCSTSp Flags (p = 0 to 63)

When the TMTRF[1:0] flag in the RSCAN0TMSTSp register is set to 10_B (transmission has been completed (without transmit abort request)) or 11_B (transmission has been completed (with transmit abort request)), the corresponding TMTCSTSp flag is set to 1.

These flags are cleared to 0 when the TMTRF[1:0] flag is set to 00_B or in channel reset mode.

Table 16.64 shows the bit allocation.

Table 16.64 TMTCSTSp Bit Allocation (1/2)

Bit Position	Channel	Transmit Buffer No.
0	0	0
1	0	1
. .	. .	. .
15	0	15
16	1	0
. .	. .	. .
30	1	14
31	1	15
32	2	0
33	2	1
. .	. .	. .

Table 16.64 TMTCSTSp Bit Allocation (2/2)

Bit Position	Channel	Transmit Buffer No.
47	2	15
48	3	0
. .	. .	. .
62	3	14
63	3	15

16.3.2.48 RSCAN0TMTASTSy — Transmit Buffer Transmit Abort Status Register (y = 0, 1)

Access: This register can be read in 8-, 16-, or 32-bit units.

Address: <RS-CAN0_base> + 0380_H + (y × 0004_H)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMTASTSp (p = y × 32 + 31)	TMTASTSp (p = y × 32 + 30)	TMTASTSp (p = y × 32 + 29)	TMTASTSp (p = y × 32 + 28)	TMTASTSp (p = y × 32 + 27)	TMTASTSp (p = y × 32 + 26)	TMTASTSp (p = y × 32 + 25)	TMTASTSp (p = y × 32 + 24)	TMTASTSp (p = y × 32 + 23)	TMTASTSp (p = y × 32 + 22)	TMTASTSp (p = y × 32 + 21)	TMTASTSp (p = y × 32 + 20)	TMTASTSp (p = y × 32 + 19)	TMTASTSp (p = y × 32 + 18)	TMTASTSp (p = y × 32 + 17)	TMTASTSp (p = y × 32 + 16)
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMTASTSp (p = y × 32 + 15)	TMTASTSp (p = y × 32 + 14)	TMTASTSp (p = y × 32 + 13)	TMTASTSp (p = y × 32 + 12)	TMTASTSp (p = y × 32 + 11)	TMTASTSp (p = y × 32 + 10)	TMTASTSp (p = y × 32 + 9)	TMTASTSp (p = y × 32 + 8)	TMTASTSp (p = y × 32 + 7)	TMTASTSp (p = y × 32 + 6)	TMTASTSp (p = y × 32 + 5)	TMTASTSp (p = y × 32 + 4)	TMTASTSp (p = y × 32 + 3)	TMTASTSp (p = y × 32 + 2)	TMTASTSp (p = y × 32 + 1)	TMTASTSp (p = y × 32 + 0)
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 16.65 RSCAN0TMTASTSy Register Contents

Bit Position	Bit Name	Function
31 to 16	TMTASTSp	Transmit Buffer Transmit Abort Status Flag p (p = y × 32 + 31 to y × 32 + 16) 0: Transmission is not aborted. 1: Transmission is aborted.
15 to 0	TMTASTSp	Transmit Buffer Transmit Abort Status Flag p (p = y × 32 + 15 to y × 32 + 0) 0: Transmission is not aborted. 1: Transmission is aborted.

TMTASTSp Flags (p = 0 to 63)

When the TMTRF[1:0] flag in the RSCAN0TMSTSp register is set to 01_B (transmit abort has been completed), the corresponding TMTASTSp flag is set to 1.

These flags are cleared to 0 when the TMTRF[1:0] flag is set to 00_B or in channel reset mode.

Table 16.66 shows the bit allocation.

Table 16.66 TMTASTSp Bit Allocation (1/2)

Bit Position	Channel	Transmit Buffer No.
0	0	0
1	0	1
.	.	.
.	.	.
15	0	15
16	1	0
.	.	.
.	.	.
30	1	14
31	1	15
32	2	0
33	2	1
.	.	.
.	.	.
47	2	15

Table 16.66 TMTASTSp Bit Allocation (2/2)

Bit Position	Channel	Transmit Buffer No.
48	3	0
. .	. .	. .
62	3	14
63	3	15

16.3.2.49 RSCAN0TMIECy — Transmit Buffer Interrupt Enable Configuration Register (y = 0, 1)

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: <RS-CAN0_base> + 0390_H + (y × 0004_H)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMIEp (p = y × 32 + 31)	TMIEp (p = y × 32 + 30)	TMIEp (p = y × 32 + 29)	TMIEp (p = y × 32 + 28)	TMIEp (p = y × 32 + 27)	TMIEp (p = y × 32 + 26)	TMIEp (p = y × 32 + 25)	TMIEp (p = y × 32 + 24)	TMIEp (p = y × 32 + 23)	TMIEp (p = y × 32 + 22)	TMIEp (p = y × 32 + 21)	TMIEp (p = y × 32 + 20)	TMIEp (p = y × 32 + 19)	TMIEp (p = y × 32 + 18)	TMIEp (p = y × 32 + 17)	TMIEp (p = y × 32 + 16)
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMIEp (p = y × 32 + 15)	TMIEp (p = y × 32 + 14)	TMIEp (p = y × 32 + 13)	TMIEp (p = y × 32 + 12)	TMIEp (p = y × 32 + 11)	TMIEp (p = y × 32 + 10)	TMIEp (p = y × 32 + 9)	TMIEp (p = y × 32 + 8)	TMIEp (p = y × 32 + 7)	TMIEp (p = y × 32 + 6)	TMIEp (p = y × 32 + 5)	TMIEp (p = y × 32 + 4)	TMIEp (p = y × 32 + 3)	TMIEp (p = y × 32 + 2)	TMIEp (p = y × 32 + 1)	TMIEp (p = y × 32 + 0)
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 16.67 RSCAN0TMIECy Register Contents

Bit Position	Bit Name	Function
31 to 16	TMIEp	Transmit Buffer Interrupt Enable bits p (p = y × 32 + 31 to y × 32 + 16) 0: Transmit buffer interrupt is disabled. 1: Transmit buffer interrupt is enabled.
15 to 0	TMIEp	Transmit Buffer Interrupt Enable bits p (p = y × 32 + 15 to y × 32 + 0) 0: Transmit buffer interrupt is disabled. 1: Transmit buffer interrupt is enabled.

TMIEp Bits (p = 0 to 63)

When any of these bits is set to 1 and the corresponding transmission has been completed, a transmit buffer interrupt request is generated.

Modify these bits when the TMTRM flag in the corresponding RSCAN0TMSTSp register is 0 (no transmit request is present).

Write 0 to the bits corresponding to the transmit buffers linked to transmit/receive FIFO buffers or the transmit buffers allocated to the transmit queue.

Table 16.68 shows the bit allocation.

Table 16.68 TMIEp Bit Allocation (1/2)

Bit Position	Channel	Transmit Buffer No.
0	0	0
1	0	1
. .	. .	. .
15	0	15
16	1	0
. .	. .	. .
30	1	14
31	1	15
32	2	0

Table 16.68 TMIEp Bit Allocation (2/2)

Bit Position	Channel	Transmit Buffer No.
33	2	1
. .	. .	. .
47	2	15
48	3	0
. .	. .	. .
62	3	14
63	3	15

16.3.2.50 RSCAN0TMIDp — Transmit Buffer ID Register (p = 0 to 63)

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: <RS-CAN0_base> + 1000_H + (p × 0010_H)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMIDE	TMRTR	THLEN	TMID[28:16]												
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMID[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 16.69 RSCAN0TMIDp Register Contents

Bit Position	Bit Name	Function
31	TMIDE	Transmit Buffer IDE 0: Standard ID 1: Extended ID
30	TMRTR	Transmit Buffer RTR 0: Data frame 1: Remote frame
29	THLEN	Transmit History Data Store Enable 0: Transmit history data is not stored in the buffer. 1: Transmit history data is stored in the buffer.
28 to 0	TMID [28:0]	Transmit Buffer ID Data Set standard ID or extended ID. For standard ID, write an ID to bits b10 to b0 and write 0 to bits b28 to b11.

Modify this register when the TMTRM bit in the corresponding RSCAN0TMSTSp register is set to 0 (no transmit request is present). If this register is linked to any transmit/receive FIFO buffer, do not write data to this register. Write data only to the transmit buffer p (p = m × 16 + 15) when this register is allocated to the transmit queue.

TMIDE Bit

This bit is used to set the ID format of the message to be transmitted from the transmit buffer.

TMRTR Bit

This bit is used to set the data format of the message to be transmitted from the transmit buffer.

THLEN Bit

Setting this bit to 1 stores a transmitted message in the transmit history buffer after transmission is completed.

TMID[28:0] Bits

These bits are used to set the ID of the message to be transmitted from the transmit buffer.

16.3.2.51 RSCAN0TMPTRp — Transmit Buffer Pointer Register (p = 0 to 63)

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: <RS-CAN0_base> + 1004_H + (p × 0010_H)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMDLC[3:0]				—	—	—	—	TMPTR[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 16.70 RSCAN0TMPTRp Register Contents

Bit Position	Bit Name	Function
31 to 28	TMDLC [3:0]	Transmit Buffer DLC Data b31 b30 b29 b28 0 0 0 0: 0 data bytes 0 0 0 1: 1 data byte 0 0 1 0: 2 data bytes 0 0 1 1: 3 data bytes 0 1 0 0: 4 data bytes 0 1 0 1: 5 data bytes 0 1 1 0: 6 data bytes 0 1 1 1: 7 data bytes 1 x x x: 8 data bytes
27 to 24	—	Reserved These bits are always read as 0. When writing, write 0.
23 to 16	TMPTR [7:0]	Transmit Buffer Label Data Set the label information to be stored in the transmit history buffer.
15 to 0	—	Reserved These bits are always read as 0. When writing, write 0.

Modify this register when the TMTRM bit in the corresponding RSCAN0TMSTSp register is set to 0 (no transmit request is present). If this register is linked to any transmit/receive FIFO buffer, do not write data to this register. Write data only to the transmit buffer p (p = m × 16 + 15) when this register is allocated to the transmit queue.

TMDLC[3:0] Bits

These bits are used to set the data length of the message to be transmitted from the transmit buffer when the TMRTR bit in the RSCAN0TMIDp register is set to 0 (data frame). If a 9-byte (or more) data length is set, 8-byte data is transmitted.

When the TMRTR bit is set to 1 (remote frame), set the data length of a requested message.

TMPTR[7:0] Bits

When message transmission has been completed, the TMPTR[7:0] value is stored in the transmit history buffer.

16.3.2.52 RSCAN0TMDF0p — Transmit Buffer Data Field 0 Register (p = 0 to 63)

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: <RS-CAN0_base> + 1008_H + (p × 0010_H)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMDB3[7:0]								TMDB2[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMDB1[7:0]								TMDB0[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 16.71 RSCAN0TMDF0p Register Contents

Bit Position	Bit Name	Function
31 to 24	TMDB3 [7:0]	Transmit Buffer Data Byte 3
23 to 16	TMDB2 [7:0]	Transmit Buffer Data Byte 2
15 to 8	TMDB1 [7:0]	Transmit Buffer Data Byte 1
7 to 0	TMDB0 [7:0]	Transmit Buffer Data Byte 0 Set transmit buffer data.

Modify this register when the TMTRM bit in the corresponding RSCAN0TMSTSp register is set to 0 (no transmit request is present). If this register is linked to any transmit/receive FIFO buffer, do not write data to this register. Write data only to the transmit buffer p (p = m × 16 + 15) when this register is allocated to the transmit queue.

16.3.2.53 RSCAN0TMDF1p — Transmit Buffer Data Field 1 Register (p = 0 to 63)

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: <RS-CAN0_base> + $100C_H$ + (p × 0010_H)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMDB7[7:0]								TMDB6[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMDB5[7:0]								TMDB4[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 16.72 RSCAN0TMDF1p Register Contents

Bit Position	Bit Name	Function
31 to 24	TMDB7 [7:0]	Transmit Buffer Data Byte 7
23 to 16	TMDB6 [7:0]	Transmit Buffer Data Byte 6
15 to 8	TMDB5 [7:0]	Transmit Buffer Data Byte 5
7 to 0	TMDB4 [7:0]	Transmit Buffer Data Byte 4 Set transmit buffer data.

Modify this register when the TMTRM bit in the corresponding RSCAN0TMSTSp register is set to 0 (no transmit request is present). If this register is linked to any transmit/receive FIFO buffer, do not write data to this register. Write data only to the transmit buffer p (p = m × 16 + 15) when this register is allocated to the transmit queue.

16.3.2.54 RSCAN0TXQCCm — Transmit Queue Configuration/Control Register (m = 0 to 3)

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: <RS-CAN0_base> + $03A0_H$ + (m × 0004_H)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	TXQIM	TXQIE	TXQDC[3:0]				—	—	—	—	—	—	—	TXQE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R/W

Table 16.73 RSCAN0TXQCCm Register Contents

Bit Position	Bit Name	Function
31 to 14	—	Reserved These bits are always read as 0. When writing, write 0.
13	TXQIM	Transmit Queue Interrupt Source Select bit 0: When the buffer becomes empty upon completion of message transmission, a transmit queue interrupt source occurs. 1: A transmit queue interrupt source occurs each time a message has been transmitted.
12	TXQIE	Transmit Queue Interrupt Enable 0: Transmit queue interrupt is disabled. 1: Transmit queue interrupt is enabled.
11 to 8	TXQDC [3:0]	Transmit Queue Depth Configuration Setting these bits to g (g = 2 to 15) makes the “g + 1” transmit queue available. Setting these bits to 0 disables the transmit queue. Setting these bits to 1 is prohibited.
7 to 1	—	Reserved These bits are always read as 0. When writing, write 0.
0	TXQE	Transmit Queue Enable 0: The transmit queue is not used. 1: The transmit queue is used.

TXQIM Bit

This bit is used to select a transmit queue interrupt source. Modify this bit in channel reset mode.

TXQIE Bit

When the TXQIE bit is set to 1 and the source selected by the TXQIM bit occurs, an interrupt request is generated.

Set the TXQE bit to 0 when modifying the TXQIE bit.

TXQDC[3:0] Bits

These bits are used to set the number of transmit buffers to be allocated to the transmit queue. Transmit buffers are allocated to the transmit queues in the descending order of buffer numbers from (m × 16 + 15) to (m × 16 + 0). For an example of buffer allocation, see **Table 16.9**. Modify these bits only in the channel reset mode.

TXQE Bit

Setting this bit to 1 makes the transmit queue available. Modify this bit in channel communication mode or channel halt mode. This bit is cleared to 0 in channel reset mode.

Set the TXQDC[3:0] bits to 0010_B or more before setting the TXQE bit to 1.

16.3.2.55 RSCAN0TXQSTSm — Transmit Queue Status Register (m = 0 to 3)

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: <RS-CAN0_base> + $03C0_H$ + (m × 0004_H)

Value after reset: $0000\ 0001_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	TXQIF	TXQFLL	TXQEMP
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W*1	R	R

Note 1. The only effective value for writing to this flag bit is 0, which clears the status. Any other writing to the bit results in retention of the status and does not change the value.

Table 16.74 RSCAN0TXQSTSm Register Contents

Bit Position	Bit Name	Function
31 to 13	—	Reserved These bits are always read as 0. When writing, write 0.
12 to 8	—	The read value is the undefined. When writing, write 0.
7 to 3	—	The read value is 0. When writing, write 0.
2	TXQIF	Transmit Queue Interrupt Request Flag 0: No transmit queue interrupt request is present. 1: A transmit queue interrupt request is present.
1	TXQFLL	Transmit Queue Full Status Flag 0: The transmit queue is not full. 1: The transmit queue is full.
0	TXQEMP	Transmit Queue Empty Status Flag 0: The transmit queue contains messages. 1: The transmit queue contains no message (transmit queue empty).

TXQIF Flag

The TXQIF flag is set to 1 when the source set by the TXQIM bit in the RSCAN0TXQCCm register has occurred.

The TXQIF flag is cleared to 0 in channel reset mode or by writing 0 to this flag. This flag is not cleared to 0 by setting the TXQE bit in the RSCAN0TXQCCm register to 0 (the transmit queue is not used).

TXQFLL Flag

The TXQFLL flag is set to 1 when the number of messages set for the transmit queue matches the transmit queue depth set by the TXQDC[3:0] bits in the RSCAN0TXQCCm register.

This flag is cleared to 0 in any of the following cases.

- The number of messages set for the transmit queue is smaller than the transmit queue depth set by the TXQDC[3:0] bits.
- In channel reset mode

TXQEMP Flag

The TXQEMP flag is cleared to 0 when even a single message is set for the transmit queue.

This flag is set to 1 in any of the following cases.

- The TXQE bit is set to 0 (the transmit queue is not used).
- The transmit queue becomes empty.
- In channel reset mode

16.3.2.56 RSCAN0TXQPCTRm — Transmit Queue Pointer Control Register (m = 0 to 3)

Access: This register can be written in 8-, 16-, or 32-bit units.

Address: <RS-CAN0_base> + $03E0_H$ + (m × 0004_H)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	TXQPC[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	W	W	W	W	W	W	W	W

Table 16.75 RSCAN0TXQPCTRm Register Contents

Bit Position	Bit Name	Function
31 to 8	—	Reserved When writing, write 0.
7 to 0	TXQPC [7:0]	Transmit Queue Pointer Control Writing FF_H to these bits moves the write pointer of the transmit queue to the next queue.

TXQPC[7:0] Bits

Writing FF_H to the TXQPC[7:0] bits moves the write pointer to the next transmit queue and generates a transmit request of the message. Write transmit messages to the RSCAN0TMIDp, RSCAN0TMPTRp, RSCAN0TMDF0p, and RSCAN0TMDF1p registers (p = m × 16 + 15) before writing FF_H to the TXQPC[7:0] bits.

Write FF_H when the TXQE bit in the RSCAN0TXQCCm register is 1 (the transmit queue is used) and the TXQFLL flag in the RSCAN0TXQSTSm register is 0 (not full).

16.3.2.57 RSCAN0THLCCm — Transmit History List Configuration/Control Register (m = 0 to 3)

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: <RS-CAN0_base> + 0400_H + (m × 0004_H)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	THLDTE	THLIM	THLIE	—	—	—	—	—	—	—	THLE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R	R	R/W

Table 16.76 RSCAN0THLCCm Register Contents

Bit Position	Bit Name	Function
31 to 11	—	Reserved These bits are always read as 0. When writing, write 0.
10	THLDTE	Transmit History Target Buffer Select 0: Entry from transmit/receive FIFO buffers and transmit queue 1: Entry from transmit buffers, transmit/receive FIFO buffers, and transmit queue
9	THLIM	Transmit History Interrupt Source Select 0: When 12 sets of data have been stored in the transmit history buffer 1: When a single set of transmit history data has been stored
8	THLIE	Transmit History Interrupt Enable 0: Transmit history interrupt is disabled. 1: Transmit history interrupt is enabled.
7 to 1	—	Reserved These bits are always read as 0. When writing, write 0.
0	THLE	Transmit History Buffer Enable 0: Transmit history buffer is not used. 1: Transmit history buffer is used.

THLDTE Bit

When this bit is set to 0, the transmit history data of messages transmitted from transmit/receive FIFO buffers and the transmit queue is stored in the transmit history buffer. When this bit is set to 1, the transmit history data of messages transmitted from transmit buffers, transmit/receive FIFO buffers, and the transmit queue is stored in the transmit history buffer.

Modify this bit only in channel reset mode.

THLIM Bit

This bit is used to select a transmit history interrupt source.

Modify this bit only in channel reset mode.

THLIE Bit

When the THLIE bit is set to 1 and the source selected by the THLIM bit has occurred, a transmit history interrupt request is generated. Modify the THLIE bit with the THLE bit set to 0.

THLE Bit

Setting this bit to 1 makes the transmit history buffer available. When data transmission from the buffer selected by the THLDTE bit has been completed, the transmit history data of the transmitted messages is stored in the transmit history buffer.

Modify this bit in channel communication mode or channel halt mode.

16.3.2.58 RSCAN0THLSTSm — Transmit History List Status Register (m = 0 to 3)

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: <RS-CAN0_base> + 0420_H + (m × 0004_H)

Value after reset: $0000\ 0001_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	THLMC[4:0]					—	—	—	—	THLIF	THLELT	THLFLL	THL EMP
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W*1	R/W*1	R	R

Note 1. The only effective value for writing to this flag bit is 0, which clears the status. Any other writing to the bit results in retention of the status and does not change the value.

Table 16.77 RSCAN0THLSTSm Register Contents

Bit Position	Bit Name	Function
31 to 13	—	Reserved These bits are always read as 0. When writing, write 0.
12 to 8	THLMC [4:0]	Transmit History Buffer Unread Data Counter These bits indicate the number of unread data sets stored in the transmit history buffer.
7 to 4	—	Reserved These bits are always read as 0. When writing, write 0.
3	THLIF	Transmit History Interrupt Request Flag 0: No transmit history interrupt request is present. 1: A transmit history interrupt request is present.
2	THLELT	Transmit History Buffer Overflow Flag 0: Transmit history buffer overflow has not occurred. 1: Transmit history buffer overflow has occurred.
1	THLFLL	Transmit History Buffer Full Status Flag 0: Transmit history buffer is not full. 1: Transmit history buffer is full.
0	THLEMP	Transmit History Buffer Empty Status Flag 0: Transmit history buffer contains unread data. 1: Transmit history buffer contains no unread data (buffer empty).

THLMC[4:0] Bits

These bits indicate the number of unread data sets stored in the transmit history buffer.

THLIF Flag

The THLIF flag is set to 1 when the interrupt source set by the THLIM bit in the RSCAN0THLCCm register has occurred.

This flag is cleared to 0 in channel reset mode or by writing 0 to this flag by the program.

THLELT Flag

The THLELT flag is set to 1 when it is attempted to store new transmit history data while the transmit history buffer is full. In this case, the new data is discarded. This flag becomes 0 in channel reset mode or by writing 0 to this flag.

THLFLL Flag

The THLFLL flag is set to 1 when 16 data sets have been stored in the transmit history buffer, and is cleared to 0 when the number of data sets stored in the transmit history buffer has decreased to less than 16. This bit is also cleared to 0 in channel reset mode or when the THLE bit in the RSCAN0THLCCm register is set to 0 (transmit history buffer is not used).

THLEMP Flag

The THLEMP flag is cleared to 0 when even a single set of transmit history data has been stored in the transmit history buffer.

This flag is set to 1 when all the data in the transmit history buffer has been read. This flag is also set to 1 in channel reset mode or when the THLE bit in the RSCAN0THLCCm register is set to 0 (transmit history buffer is not used).

NOTE

To clear the THLIF and THLELT flags to 0, the program must write 0 to the corresponding flag to be cleared. When writing 0, using a store instruction, set 1 to other flags. Note that when using a bit-manipulation instruction for writing to this register, the read-modify-write operation may inadvertently clear the flags.

16.3.2.59 RSCAN0THLACCm — Transmit History List Access Register (m = 0 to 3)

Access: This register can be read in 8-, 16-, or 32-bit units.

Address: <RS-CAN0_base> + 1800_H + (m × 0004_H)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TID[7:0]								—	BN[3:0]				BT[2:0]		
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 16.78 RSCAN0THLACCm Register Contents

Bit Position	Bit Name	Function
31 to 16	—	Reserved These bits are always read as 0.
15 to 8	TID[7:0]	Label Data The label information of stored data can be read.
7	—	Reserved This bit is always read as 0. When writing, write 0.
6 to 3	BN[3:0]	Buffer Number Data The buffer number of transmit source (transmit buffer, transmit/receive FIFO or transmit queue) can be read.
2 to 0	BT[2:0]	Buffer Type Data b2 b1 b0 0 0 1: Transmit buffer 0 1 0: Transmit/receive FIFO buffer 1 0 0: Transmit queue

TID[7:0] Bits

These bits indicate the label information of transmit history data stored in the transmit history buffer.

BN[3:0] Bits

These bits indicate the transmit source buffer number of transmit history data stored in the transmit history buffer.

BT[2:0] Bits

These bits indicate the transmit source buffer type of transmit history data stored in the transmit history buffer.

16.3.2.60 RSCAN0THLPCTRm — Transmit History List Pointer Control Register (m = 0 to 3)

Access: This register can be written in 8-, 16-, or 32-bit units.

Address: <RS-CAN0_base> + 0440_H + (m × 0004_H)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	THLPC[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	W	W	W	W	W	W	W	W

Table 16.79 RSCAN0THLPCTRm Register Contents

Bit Position	Bit Name	Function
31 to 8	—	Reserved When writing, write 0.
7 to 0	THLPC [7:0]	Transmit History List Pointer Control Writing FF_H to these bits moves the read pointer to the next unread data in the transmit history buffer.

THLPC[7:0] Bits

When the THLPC[7:0] bits are set to FF_H, the read pointer moves to the next data in the transmit history buffer. At this time, the THLMC[4:0] (transmit history buffer unread data counter) value in the RSCAN0THLSTSm register is decremented by 1. After reading the RSCAN0THLACCm register, write FF_H to the THLPC[7:0] bits.

Write FF_H to the THLPC[7:0] bits when the THLE bit in the RSCAN0THLCCm register is set to 1 (transmit history buffer is used) and the THLEMP flag in the RSCAN0THLSTSm register is 0.

16.3.2.61 RSCAN0GTSTCFG — Global Test Configuration Register

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: <RS-CAN0_base> + 0468_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	RTMPS[6:0]						
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	C3ICBCE	C2ICBCE	C1ICBCE	C0ICBCE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Table 16.80 RSCAN0GTSTCFG Register Contents

Bit Position	Bit Name	Function
31 to 23	—	Reserved These bits are always read as 0. When writing, write 0.
22 to 16	RTMPS [6:0]	RAM Test Page Configuration Set a value within a range of page 0 (00_H) to page 47 ($2F_H$).
15 to 4	—	Reserved These bits are always read as 0. When writing, write 0.
3	C3ICBCE	CAN3 Inter-Channel Communication Test Enable 0: CAN3 inter-channel communication test is disabled. 1: CAN3 inter-channel communication test is enabled.
2	C2ICBCE	CAN2 Inter-Channel Communication Test Enable 0: CAN2 inter-channel communication test is disabled. 1: CAN2 inter-channel communication test is enabled.
1	C1ICBCE	CAN1 Inter-Channel Communication Test Enable 0: CAN1 inter-channel communication test is disabled. 1: CAN1 inter-channel communication test is enabled.
0	C0ICBCE	CAN0 Inter-Channel Communication Test Enable 0: CAN0 inter-channel communication test is disabled. 1: CAN0 inter-channel communication test is enabled.

Modify the RSCAN0GTSTCFG register in global test mode.

RTMPS[6:0] Bits

These bits are used to set the RAM test target page number for RAM test. Set a value from 00_H to $2F_H$.

C3ICBCE Bit

Setting this bit to 1 enables channel 3 inter-channel communication test.

C2ICBCE Bit

Setting this bit to 1 enables channel 2 inter-channel communication test.

C1ICBCE Bit

Setting this bit to 1 enables channel 1 inter-channel communication test.

C0ICBCE Bit

Setting this bit to 1 enables channel 0 inter-channel communication test.

16.3.2.62 RSCAN0GTSTCTR — Global Test Control Register

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: <RS-CAN0_base> + $046C_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	RTME	—	ICBCTME
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R	R/W

Table 16.81 RSCAN0GTSTCTR Register Contents

Bit Position	Bit Name	Function
31 to 3	—	Reserved These bits are always read as 0. When writing, write 0.
2	RTME	RAM Test Enable 0: RAM test is disabled. 1: RAM test is enabled.
1	—	Reserved These bits are always read as 0. When writing, write 0.
0	ICBCTME	Inter-Channel Communication Test Enable 0: Inter-channel communication test is disabled. 1: Inter-channel communication test is enabled.

RTME Bit

Setting this bit to 1 enables RAM test. Modify this bit in global test mode.

1. Set the GMDC[1:0] bits in the RSCAN0GCTR register to 10_B (global test mode).
2. Set the RTME bit to 1.
3. Check that the RTME bit is set to 1.

ICBCTME Bit

When this bit is set to 1, a communication test is enabled between the channels for which the CmICBCE bit (m = 0 to 3) in the RSCAN0GTSTCFG register has been set to 1. Modify the ICBCTME bit only in global test mode.

16.3.2.63 RSCAN0GLOCKK — Global Lock Key Register

Access: This register can be written in 16, 32 bits/unit

Address: <RS-CAN0_base> + $047C_H$

Value after reset: $0000\ 0000_H$.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LOCK[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1	W*1

Note 1. Writing to these bits is effective only when the RS-CAN module is in global test mode.

Table 16.82 RSCAN0GLOCKK Register Contents

Bit Position	Bit Name	Function
31 to 16	—	Reserved These bits are read as an undefined value. When writing, write 0.
15 to 0	LOCK[15:0]	Lock Key These bits are key bits to release protection of test mode.

The RSCAN0GLOCKK register is a write-only register to release protection of special test bits. For protection released data, **Section 16.5.4.2, Procedure for Releasing the Protection**.

LOCK[15:0] Bits

Writing the protection release data to the LOCK[15:0] bits in succession enables writing 1 to the RTME bit in the GTSTCTR register.

After the protection has been released, writing to the I/O register area of CAN (<RS-CAN0_base> + 0000_H to <RS-CAN0_base> + $04FF_H$), excluding the RAM, enables the protection again.

Reading from the register of the CAN I/O register area or reading from/wring to other areas does not enable the protection.

16.3.2.64 RSCAN0RPGACCr — RAM Test Page Access Register (r = 0 to 63)

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: <RS-CAN0_base> + 1900_H + (r × 0004_H)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RDTA[31:16]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RDTA[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 16.83 RSCAN0RPGACCr Register Contents

Bit Position	Bit Name	Function
31 to 0	RDTA [31:0]	RAM Data Test Access Data can be read and written in CAN RAM.

Modify the RSCAN0RPGACCr register in global test mode with the RTME bit in the RSCAN0GTSTCTR register set to 1 (RAM test is enabled).

The RSCAN0RPGACCr register is readable and writable when the RTME bit is set to 1.

16.4 Functions

16.4.1 Interrupt Sources

The RS-CAN module has 14 interrupts that are grouped into global interrupts and channel interrupts.

- Global interrupts (2 interrupts):
  - Receive FIFO interrupt
  - Global error interrupt
- Channel interrupts (3 interrupts per channel):
  CANm transmission interrupt (m = 0 to 3)
  - CANm transmission complete interrupt
  - CANm transmission abort interrupt
  - CANm transmit/receive FIFO transmission complete interrupt (in transmit mode or gateway mode)
  - CANm transmit history interrupt
  - CANm transmit queue interrupt

  CANm transmit/receive FIFO reception complete interrupt (in receive mode or gateway mode)

  CANm error interrupt

When an interrupt request is generated, the corresponding interrupt request flag is set to 1 (interrupt request present). In that case, when the interrupt enable bit is 1 (enabling interrupts), an interrupt request is output from the RS-CAN module. (Generation of interrupts also depends on the interrupt control register settings of the interrupt controller.)

Setting the interrupt request flag to 0 (no interrupt request present) or setting the interrupt enable bit to 0 (disabling interrupts) clears the interrupt request. The interrupt request is kept being output until the interrupt request flag is cleared.

Table 16.84 lists the CAN interrupt sources. **Figure 16.2** shows the CAN global interrupt block diagram, and **Figure 16.3** shows the CAN channel interrupt block diagram.

Table 16.84 List of CAN Interrupt Sources

		Interrupt Source	Corresponding Interrupt Request Flag	Corresponding Interrupt Enable Bit
Global interrupts	Receive FIFO	Receive FIFO 0	RFIF flag in the RSCAN0RFSTS0 register	RFIE bit in the RSCAN0RFCC0 register
		Receive FIFO 1	RFIF flag in the RSCAN0RFSTS1 register	RFIE bit in the RSCAN0RFCC1 register
		Receive FIFO 2	RFIF flag in the RSCAN0RFSTS2 register	RFIE bit in the RSCAN0RFCC2 register
		Receive FIFO 3	RFIF flag in the RSCAN0RFSTS3 register	RFIE bit in the RSCAN0RFCC3 register
		Receive FIFO 4	RFIF flag in the RSCAN0RFSTS4 register	RFIE bit in the RSCAN0RFCC4 register
		Receive FIFO 5	RFIF flag in the RSCAN0RFSTS5 register	RFIE bit in the RSCAN0RFCC5 register
		Receive FIFO 6	RFIF flag in the RSCAN0RFSTS6 register	RFIE bit in the RSCAN0RFCC6 register
		Receive FIFO 7	RFIF flag in the RSCAN0RFSTS7 register	RFIE bit in the RSCAN0RFCC7 register
	Global error		DEF flag in the RSCAN0GERFL register MES flag in the RSCAN0GERFL register THLES flag in the RSCAN0GERFL register	DEIE bit in the RSCAN0GCTR register MEIE bit in the RSCAN0GCTR register THLEIE bit in the RSCAN0GCTR register
Channel interrupts (m = 0 to 3)	CANm transmit	CANm transmit complete	TMTRF[1:0] flag in the RSCAN0TMSTSp register	TMIE bit in the RSCAN0TMIECy register
		CANm transmit abort	TMTRF[1:0] flag in the RSCAN0TMSTSp register	TAIE bit in the RSCAN0CmCTR register
		CANm transmit/ receive FIFO transmit complete	CFTXIF flag in the RSCAN0CFSTSk register	CFTXIE bit in the RSCAN0CFCCk register
		CANm transmit queue	TXQIF flag in the RSCAN0TXQSTSm register	TXQIE bit in the RSCAN0TXQCCm register
		CANm transmit history	THLIF flag in the RSCAN0THLSTSm register	THLIE bit in the RSCAN0THLCCm register
	CANm transmit/receive FIFO reception complete		CFRXIF flag in the RSCAN0CFSTSk register	CFRXIE bit in the RSCAN0CFCCk register
	CANm error		• BEF flag in the RSCAN0CmERFL register • ALF flag in the RSCAN0CmERFL register • BLF flag in the RSCAN0CmERFL register • OVLF flag in the RSCAN0CmERFL register • BORF flag in the RSCAN0CmERFL register • BOEF flag in the RSCAN0CmERFL register • EPF flag in the RSCAN0CmERFL register • EWF flag in the RSCAN0CmERFL register	• BEIE bit in the RSCAN0CmCTR register • ALIE bit in the RSCAN0CmCTR register • BLIE bit in the RSCAN0CmCTR register • OLIE bit in the RSCAN0CmCTR register • BORIE bit in the RSCAN0CmCTR register • BOEIE bit in the RSCAN0CmCTR register • EPIE bit in the RSCAN0CmCTR register • EWIE bit in the RSCAN0CmCTR register

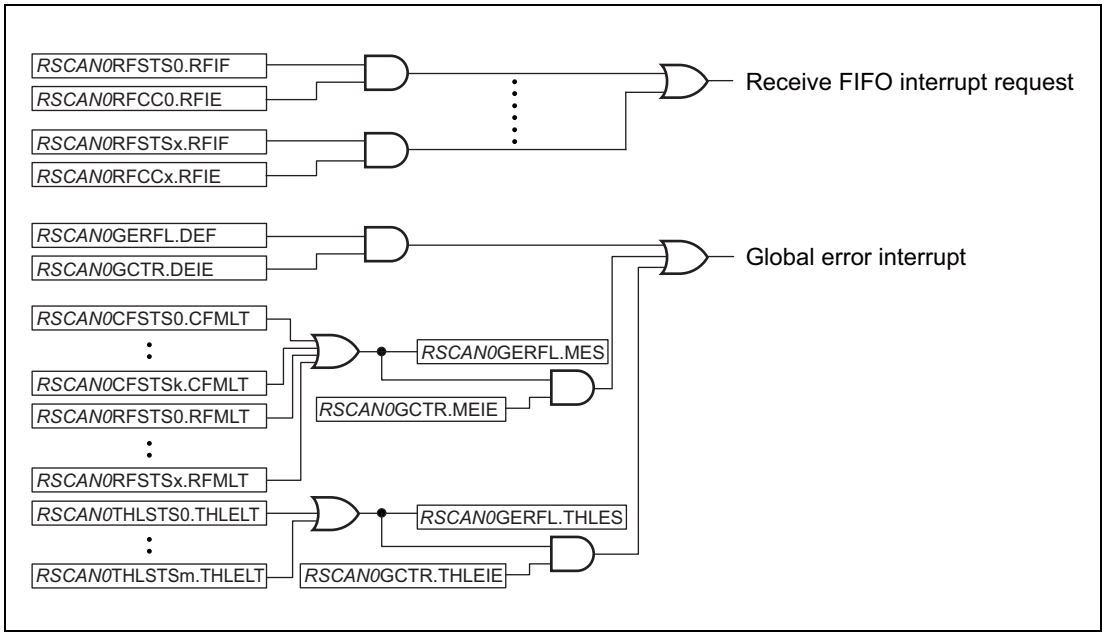


Figure 16.2 CAN Global Interrupt Block Diagram

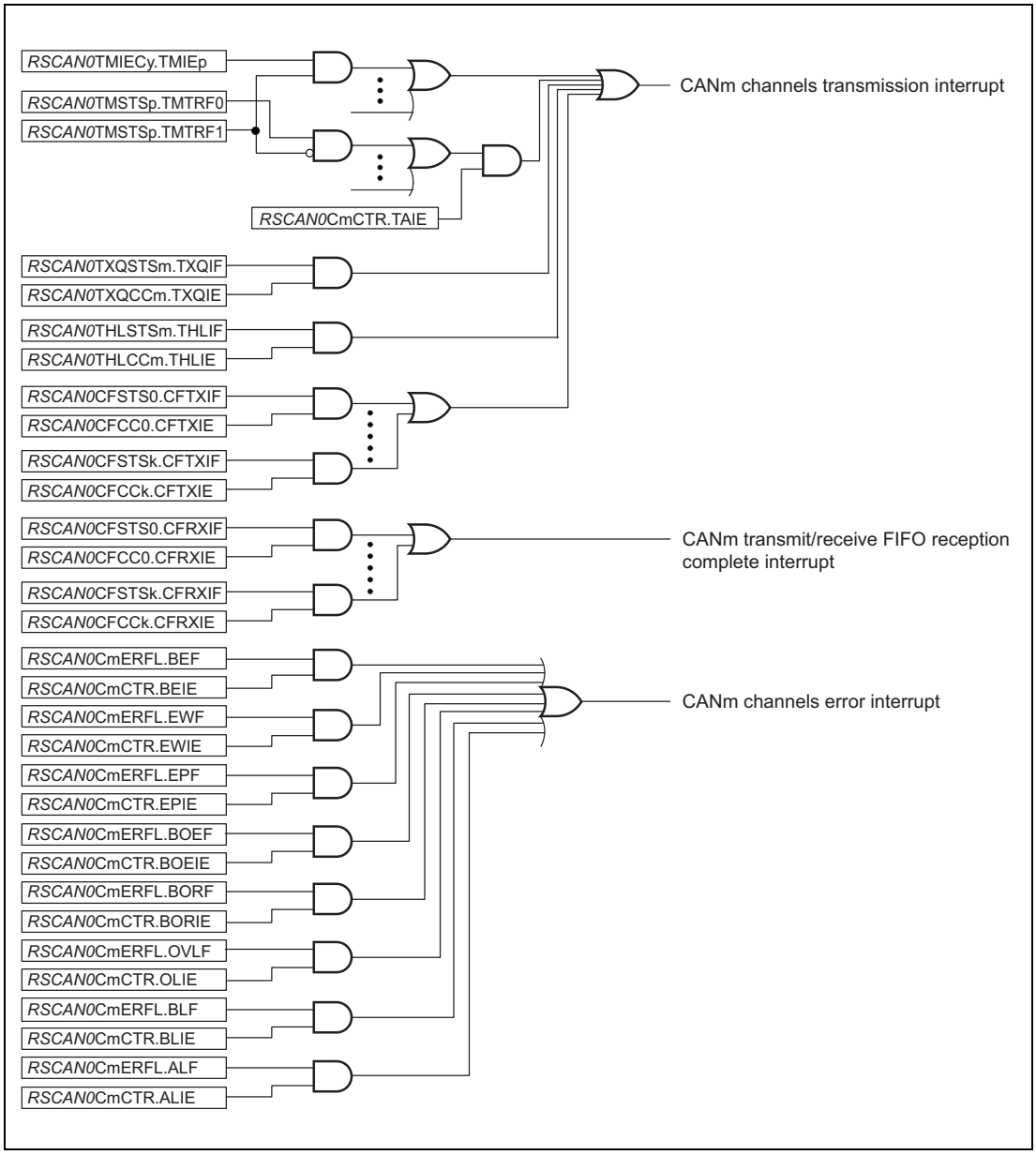


Figure 16.3 CAN Channel Interrupt Block Diagram

16.4.2 CAN Modes

The RS-CAN module has four global modes to control entire RS-CAN module status and four channel modes to control individual channel status. Details of global modes are described in **Section 16.4.2.1, Global Modes**, and details of channel modes are described in **Section 16.4.2.6, Channel Modes**.

- Global stop mode: Stops clocks of the entire module to achieve low power consumption.
- Global reset mode: Performs initial settings for the entire module.
- Global test mode: Performs test settings and performs RAM test.
- Global operating mode: Makes the entire module operable.
- Channel stop mode: Stops the channel clock.
- Channel reset mode: Performs initial settings for the channel.
- Channel halt mode: Stops CAN communication and performs channel test.
- Channel communication mode: Performs CAN communication.

16.4.2.1 Global Modes

Figure 16.4 shows the transitions of global modes.

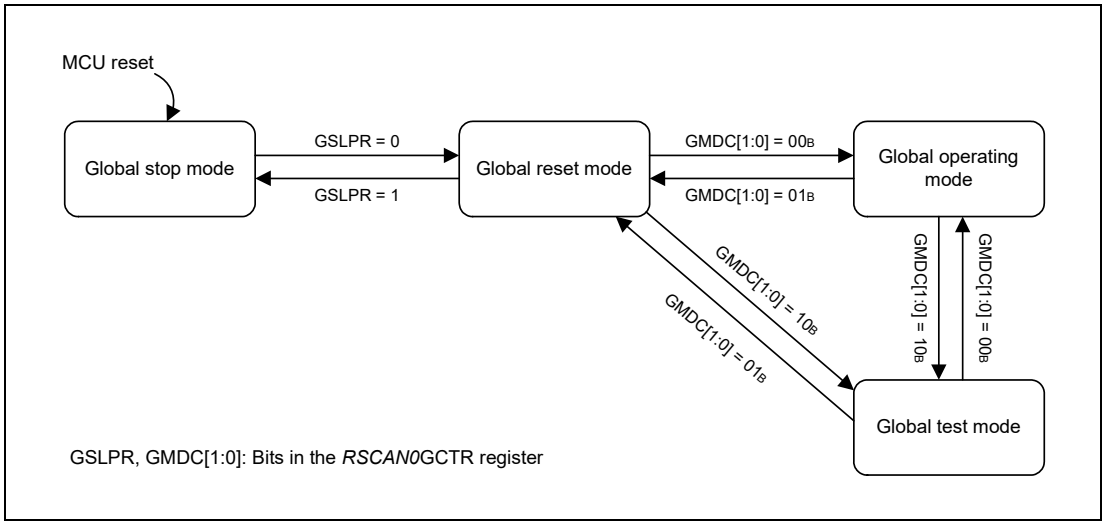


Figure 16.4 Transitions of Global Modes

Channel modes may transition in accordance with transitions of global modes. **Table 16.85** shows the transitions of channel modes depending on the global mode setting (GMDC[1:0] bits and GSLPR bit).

Table 16.85 Transitions of Channel Modes Depending on Global Mode Setting (GMDC[1:0] and GSLPR Bits)

Channel Mode before Setting	Channel Mode after Setting			
	GMDC[1:0] = 00_B GSLPR = 0 (Global Operating)	GMDC[1:0] = 10_B GSLPR = 0 (Global Test)	GMDC[1:0] = 01_B GSLPR = 0 (Global Reset)	GMDC[1:0] = 01_B GSLPR = 1 (Global Stop)
Channel communication	Channel communication	Channel halt	Channel reset	Transition prohibited
Channel halt	Channel halt	Channel halt	Channel reset	Transition prohibited
Channel reset	Channel reset	Channel reset	Channel reset	Channel stop
Channel stop	Channel stop	Channel stop	Channel stop	Channel stop

Note: GMDC[1:0], GSLPR: Bits in the RSCAN0GCTR register

Table 16.86 shows the transition time between each global mode.

Table 16.86 Transition Time of Global Mode

Mode before Transition	Mode after Transition	Maximum Transition Time
Global stop	Global reset	3 clocks of pclk
Global reset	Global stop	3 clocks of pclk
Global reset	Global test	10 clocks of pclk
Global reset	Global operation	10 clocks of pclk
Global test	Global reset	3 clocks of pclk
Global test	Global operation	3 clocks of pclk
Global operation	Global reset	3 clocks of pclk
Global operation	Global test	2 times of CAN frame*1

Note 1. CAN frame time of the channel whose communication speed is lowest among the channels in use.

16.4.2.2 Global Stop Mode

In global stop mode, the CAN clocks do not run and therefore power consumption is reduced. CAN registers can be read, but writing data to them is prohibited. Register values are retained. In global stop mode, only the clock for CPU writing to the GSLPR bit is running.

After the MCU is reset, the RS-CAN module transitions to global stop mode. Setting the GSLPR bit in the RSCAN0GCTR register to 1 (in global stop mode) in global reset mode sets the CSLPR bit in each of the RSCAN0CmCTR register to 1 (channel stop mode). If all channels are forcibly caused to transition to channel stop mode, the RS-CAN module transitions to global stop mode. The GSLPR bit should not be modified in global operating mode and global test mode.

16.4.2.3 Global Reset Mode

In global reset mode, RS-CAN module settings are performed. When the RS-CAN module transitions to global reset mode, some registers are initialized. **Table 16.89** and **Table 16.90** list the registers to be initialized.

Setting the GMDC[1:0] bits in the RSCAN0GCTR register to 01_B sets the CHMDC[1:0] bits in each of the RSCAN0CmCTR register (m = 0 to 3) to 01_B (channel reset mode). If all channels are forcibly caused to transition to channel reset mode, the RS-CAN module transitions to global reset mode. Channels that are already in channel reset mode or channel stop mode do not transition (because the CHMDC[1:0] bits have already been set to 01_B).

16.4.2.4 Global Test Mode

In global test mode, settings for test-related registers are performed. When the RS-CAN module transitions to global test mode, all CAN communications are disabled.

Setting the GMDC[1:0] bits in the RSCAN0GCTR register to 10_B sets the CHMDC[1:0] bits in each of the RSCAN0CmCTR register to 10_B (channel halt mode). If all channels are forcibly caused to transition to channel halt mode, the RS-CAN module transitions to global test mode. Channels that are in channel stop mode, channel reset mode, or channel halt mode do not transition.

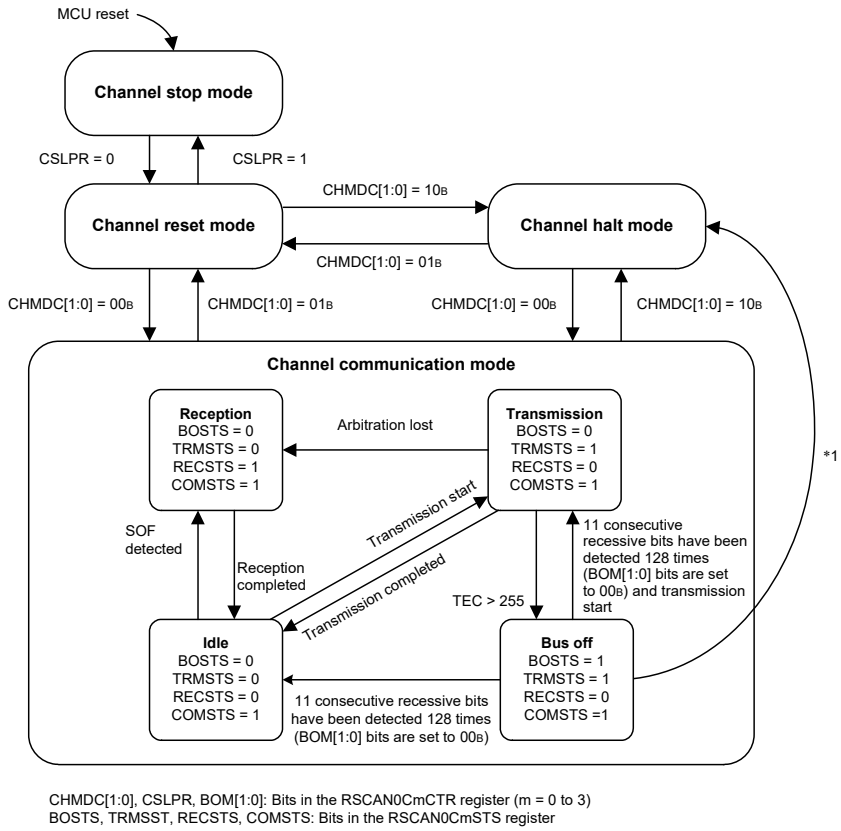
16.4.2.5 Global Operating Mode

In global operating mode, entire RS-CAN module operates.

When the GMDC[1:0] bits in the RSCAN0GCTR register are set to 00_B, the RS-CAN module transitions to global operating mode.

16.4.2.6 Channel Modes

Figure 16.5 shows a channel mode state transition chart. **Table 16.87** shows transition time between channel modes.



Note 1. Timing of transition from bus off state to channel halt mode
When BOM[1:0] = 01_B: Transition to channel halt CHMDC when TEC exceeds 255
When BOM[1:0] = 10_B: Transition to channel halt CHMDC when 11 consecutive recessive bits have been detected 128 times
When BOM[1:0] = 11_B: Transition to channel halt CHMDC when the CHMDC[1:0] bits are set to 10_B

Figure 16.5 Channel Mode State Transition Chart

Table 16.87 Transition Time of Channel Mode (m = 0 to 3)

Mode before Transition	Mode after Transition	Maximum Transition Time
Channel stop	Channel reset	3 clocks of pclk
Channel reset	Channel stop	3 clocks of pclk
Channel reset	Channel Halt	3 CANm bit time
Channel reset	Channel communication	2 CANm bit time
Channel Halt	Channel reset	3 clocks of pclk
Channel Halt	Channel communication	3 CANm bit time
Channel communication	Channel reset	3 clocks of pclk
Channel communication	Channel Halt	2 times of CANm frame

16.4.2.7 Channel Stop Mode

In channel stop mode, clocks are not supplied to channels and therefore power consumption is reduced. CAN registers can be read, but writing data to them is prohibited. Register values are retained.

Each channel enters channel stop mode after the MCU is reset. The channel transitions to channel stop mode when the CSLPR bit in the RSCAN0CmCTR register (m = 0 to 3) is set to 1 (channel stop mode) in channel reset mode. The CSLPR bit should not be modified in channel communication mode and channel halt mode.

16.4.2.8 Channel Reset Mode

In channel reset mode, channel settings are performed. When a channel transitions to channel reset mode, some channel-related registers are initialized. **Table 16.89** lists the registers to be initialized.

When the CHMDC[1:0] bits in the RSCAN0CmCTR register are set to 01_B (channel reset mode) during CAN communication, communication is terminated before it is completed and the channel transitions to channel reset mode. **Table 16.88** shows the operation when the CHMDC[1:0] bits are set to 01_B (channel reset mode) during CAN communication.

16.4.2.9 Channel Halt Mode

In channel halt mode, settings for test-related registers of channels are performed. When a channel transitions to channel halt mode, CAN communication of the channel stops.

Table 16.88 shows operation when the CHMDC[1:0] bits are set to 10_B (channel halt mode) during CAN communication.

Table 16.88 Operation When a Channel Transitions to Channel Reset Mode/Channel Halt Mode

Mode	During Reception	During Transmission	Bus Off State
Channel reset (CHMDC[1:0] = "01_B")	Transitions to channel reset mode before reception is completed.*1	Transitions to channel reset mode before transmission is completed.*1	Transitions to channel reset mode before bus off recovery is completed.
Channel halt*3 (CHMDC[1:0] = "10_B")	Transitions to channel halt mode after reception is completed.*2	Transitions to channel halt mode after transmission is completed.	[When BOM[1:0] = 00_B] Transitions to channel halt mode (CHMDC[1:0] = 10_B) only after bus off recovery. [When BOM[1:0] = 01_B] Transitions to channel halt mode automatically when the condition for transition to bus off state is met. [When BOM[1:0] = 10_B] Transitions to channel halt mode automatically after bus off recovery is completed. [When BOM[1:0] = 11_B] Transitions to channel halt mode immediately after the CHMDC[1:0] bits are set to 10_B before bus off recovery is completed.

Note 1. To allow transition to channel reset mode after communication is completed, set the CHMDC[1:0] bits to 10_B and confirm that communication has been completed and transition to channel halt mode has been made, and then set the CHMDC[1:0] bits to 01_B.

Note 2. While the CAN bus is locked at the dominant level, transition to channel halt mode is not made. In that case, enter channel reset mode. The CAN bus status can be confirmed with the BLF flag of the RSCAN0CmERFL register that becomes 1 when dominant lock is detected.

Note 3. When the transition from channel reset mode to channel wait mode is to be made, set the RSCAN0CmBCFG register in channel reset mode and then shift to channel wait RSCAN0CmCFG.

16.4.2.10 Channel Communication Mode

In channel communication mode, CAN communication is performed. Each channel is in the following communication states during CAN communication.

- Idle : Neither reception nor transmission is in progress.
- Reception : Receiving a message sent from another node.
- Transmission : Transmitting a message.
- Bus off : Isolated from CAN communication.

When the CHMDC[1:0] bits in the RSCAN0CmCTR register are set to 00_B, the channel transitions to channel communication mode. After that, when 11 consecutive recessive bits have been detected, the COMSTS flag in the RSCAN0CmSTS register (m = 0 to 3) is set to 1 (communication is ready) and transmission and reception are enabled on the CAN network as an active node. At this time, transmission and reception of messages can be started.

16.4.2.11 Bus Off State

A channel transitions to the bus off state according to the transmit/receive error counter increment/ decrement rules of the CAN specifications. How to return from the bus off state is set by the BOM[1:0] bits in the RSCAN0CmCTR register.

- When BOM[1:0] = 00_B:
  Bus off recovery is compliant with the CAN specification. After 11 consecutive recessive bits have been detected 128 times, a channel returns from the bus off state to the CAN communication ready state (error active state). At that time, the TEC[7:0] and REC[7:0] bits in the RSCAN0CmSTS register are initialized to 00_H , the BORF flag in the RSCAN0CmERFL register is set to 1 (bus off recovery is detected), and a bus off recovery interrupt request is generated. When the CHMDC[1:0] bits in the RSCAN0CmCTR register are set to 10_B (channel halt mode) in the bus off state, the channel transitions to channel halt mode after completion of bus off recovery (11 consecutive recessive bits have been detected 128 times).
- When BOM[1:0] = 01_B:
  When a channel transitions to the bus off state, the CHMDC[1:0] bits are set to 10_B and the channel transitions to channel halt mode. At that time, the TEC[7:0] and REC[7:0] bits are initialized to 00_H but the BORF flag is not set to 1 and a bus off recovery interrupt request is not generated.
- When BOM[1:0] = 10_B:
  When a channel has transitioned to the bus off state, the CHMDC[1:0] bits are set to 10_B. After completion of bus off recovery (11 consecutive recessive bits have been detected 128 times), the channel transitions to channel halt mode. At that time, the TEC[7:0] and REC[7:0] bits are initialized to 00_H and the BORF flag is set to 1, and bus off recovery interrupt request is generated.
- When BOM[1:0] = 11_B:
  When the CHMDC[1:0] bits are set to 10_B in the bus off state, the channel transitions to channel halt mode before bus off recovery is completed. At that time, the TEC[7:0] and REC[7:0] bits are initialized to 00_H but the BORF flag is not set to 1. And bus off recovery interrupt request is not generated.
  However, the BORF flag becomes 1, and bus off recovery interrupt request is generated if a RS-CAN module transitions to error active state (by detecting 128 times of 11 consecutive recessive bits) before CHMDC[1:0] bits are set to 10_B.

If the channel transitions to channel halt mode by the RS-CAN module simultaneously when the program writes a value to the CHMDC[1:0] bits, writing by the program takes precedence. An automatic transition to channel halt mode when the BOM[1:0] bits are set to 01_B or 10_B is made only when the CHMDC[1:0] bits are 00_B (channel communication mode).

Furthermore, setting the RTBO bit in the RSCAN0CmCTR register to 1 allows forcible return from the bus off state. As soon as the RTBO bit is set to 1, the state changes to the error active state. After 11 consecutive recessive bits have been detected, the RS-CAN module becomes ready for communication. In this case, the BORF flag is not set to 1 and the TEC[7:0] and REC[7:0] bits are initialized to 00_H. Write 1 to the RTBO bit when the BOM[1:0] value is 00_B. If you write 1 to the RTBO bit in any other cases than the buss off state, writing is ignored and the RTBO bit becomes 0 immediately.

Table 16.89 Registers Initialized in Global Reset Mode or Channel Reset Mode

Register	Bit / Flag
RSCAN0CmCTR register	CTMS[1:0], CTME, CHMDC[1:0]
RSCAN0CmSTS register	CHLTSTS, EPSTS, BOSTS, TRMSTS, RECSTS, COMSTS, REC[7:0], TEC[7:0]
RSCAN0CmERFL register	CRCREG[14:0], ADERR, B0ERR, B1ERR, CERR, AERR, FERR, SERR, ALF, BLF, OVLF, BORF, BOEF, EPF, EWF, BEF
RSCAN0CFCCk register	When transmit/receive FIFO buffer is in transmit mode or gateway mode: CFE
RSCAN0CFSTSk register	When transmit/receive FIFO buffer is in transmit mode or gateway mode: CFMC[7:0], CFFLL, CFEMP, CFMLT, CFRXIF, CFTXIF
RSCAN0CFTISTS register	CFkTXIF
RSCAN0TMCp register	TMOM, TMTAR, TMTR
RSCAN0TMSTSp register	TMTARM, TMTRM, TMTRF[1:0], TMTSTS
RSCAN0TMTRSTSy register	TMTRSTSp (Bits of corresponding channel are initialized in channel reset mode.)
RSCAN0TMTARSTSy register	TMTARSTSp (Bits of corresponding channel are initialized in channel reset mode.)
RSCAN0TMTCSTSy register	TMTCSTSp (Bits of corresponding channel are initialized in channel reset mode.)
RSCAN0TMTASTSy register	TMTASTSp (Bits of corresponding channel are initialized in channel reset mode.)
RSCAN0TXQCCm register	TXQE
RSCAN0TXQSTSm register	TXQIF, TXQFLL, TXQEMP
RSCAN0THLCCm register	THLE
RSCAN0THLSTSm register	THLMC[4:0], THLIF, THLELT, THLFLL,THLEMP
RSCAN0GTINTSTS0 register	TSIFm, TAIFm, TQIFm, CFTIFm, THIFm (m = 0 to 3)

Table 16.90 Registers Initialized Only in Global Reset Mode

Register	Bit / Flag
RSCAN0GSTS register	GHLTSTS
RSCAN0GERFL register	THLES, MES, DEF
RSCAN0GTSC register	TS[15:0]
RSCAN0RMNDy register	RMNSq
RSCAN0RFCCx register	RFE
RSCAN0RFSTSx register	RFMC[7:0], RFIF, RFMLT, RFFLL, RFEMP
RSCAN0CFCCk register	When transmit/receive FIFO buffer is in receive mode: CFE
RSCAN0CFSTSk register	When transmit/receive FIFO buffer is in receive mode: CFMC[7:0], CFFLL, CFEMP, CFTXIF, CFRXIF, CFMLT
RSCAN0FESTS register	CFkEMP, RFxEMP
RSCAN0FFSTS register	CFkFLL, RFxFLL
RSCAN0FMSTS register	CFkMLT, RFxMLT
RSCAN0RFISTS register	RFxIF
RSCAN0CFRISTS register	CFkRXIF
RSCAN0GTSTCFG register	RTMPS[6:0], C0ICBCE, C1ICBCE, C2ICBCE, C3ICBCE
RSCAN0GTSTCTR register	RTME, ICBCTME

16.4.3 Function for Reception

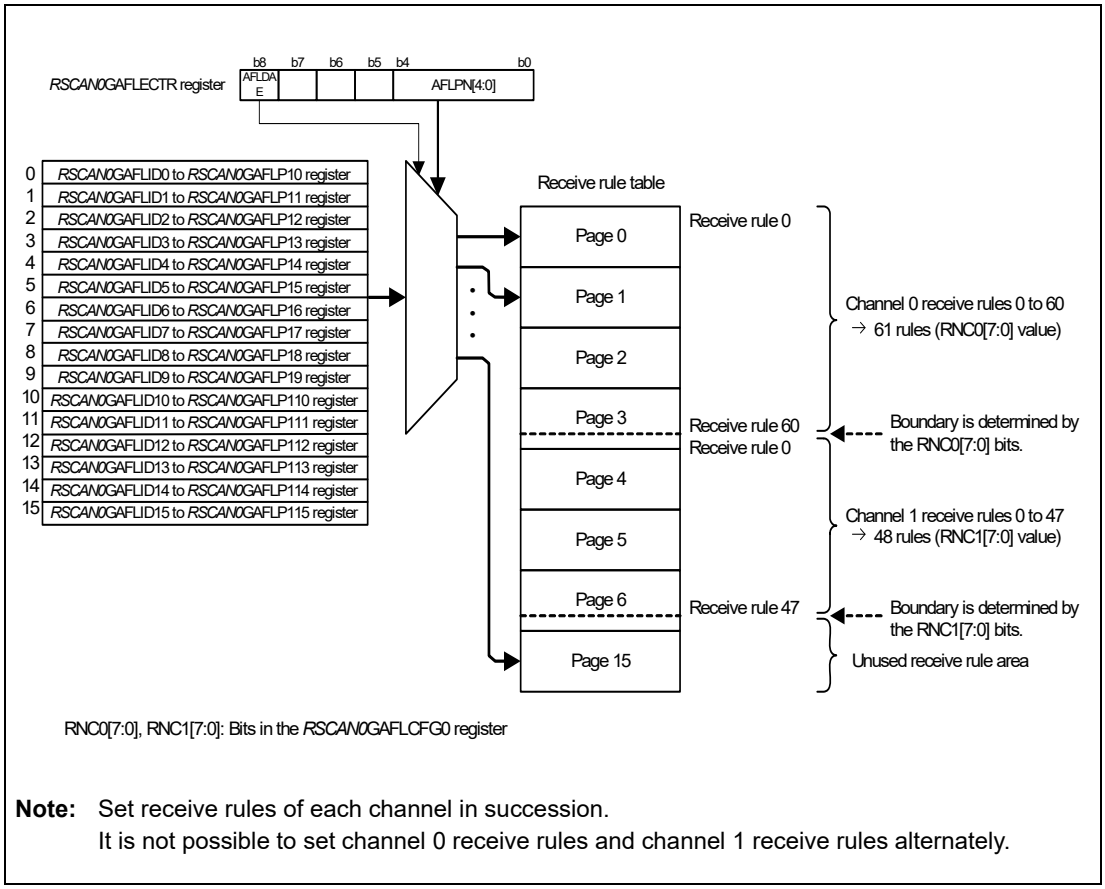
There are two reception types.

- Reception by receive buffers:
  Zero to 64 receive buffers can be shared by all channels. Since messages stored in receive buffers are overwritten at each reception, the latest received data can always be read.
- Reception by receive FIFO buffers and transmit/receive FIFO buffers (receive mode):
  Eight receive FIFO buffers can be shared by all channels and three dedicated transmit/receive FIFO buffers are provided for each channel. Messages of the number specified by the buffer depth can be stored in FIFO buffers and can be read sequentially from the oldest.

16.4.3.1 Data Processing Using the Receive Rule Table

Data processing using the receive rule table allows selected messages to be stored in the specified buffer. Data processing includes acceptance filter processing, DLC filter processing, routing processing, label addition processing, and mirror function processing.

Up to 128 receive rules can be registered per channel and up to “64 × number of channels” receive rules can be registered in the entire module. (Up to 256 receive rules can be registered in this module that has four channels.) Set receive rules for each channel. No receive rule can be shared with other channels. If receive rules are not set, no message can be received. **Figure 16.6** illustrates how receive rules are registered, giving an example using 2 channels of RS-CAN.



Note: Set receive rules of each channel in succession.
It is not possible to set channel 0 receive rules and channel 1 receive rules alternately.

Figure 16.6 Entry of Receive Rules (in the case of setting channel 0,1)

Each receive rule consists of 16 bytes of the RSCAN0GAFLIDj, RSCAN0GAFLMj, RSCAN0GAFLP0j, and RSCAN0GAFLP1j registers (j = 0 to 15). The RSCAN0GAFLIDj register is used to set GAFLID, GAFLIDE bit, GAFLRTR bit, and the mirror function. The RSCAN0GAFLMj register is used to set mask. The RSCAN0GAFLP0j register is used to set label information to be added, DLC value, and destination receive buffer. The RSCAN0GAFLP1j register is used to set a destination FIFO buffer. Up to 16 receive rules can be set per page.

16.4.3.2 Acceptance Filter Processing

In the acceptance filter processing, the ID data, IDE bit, and RTR bit in a received message are compared with the ID data, IDE bit, and RTR bit set in the receive rule of the corresponding channel. When all these bits match, the message passes through the acceptance filter processing. Bits set to 0 (bits are not compared) in the RSCAN0GAFLMj register are not compared and are regarded as matched.

Check begins with the lowest-numbered receive rule. When all the bits to be compared in a received message match the bits set in the receive rule or when all the receive rules are compared without any match, filter processing stops. If there is no matching receive rule, the received message is not stored in the receive buffer or FIFO buffer.

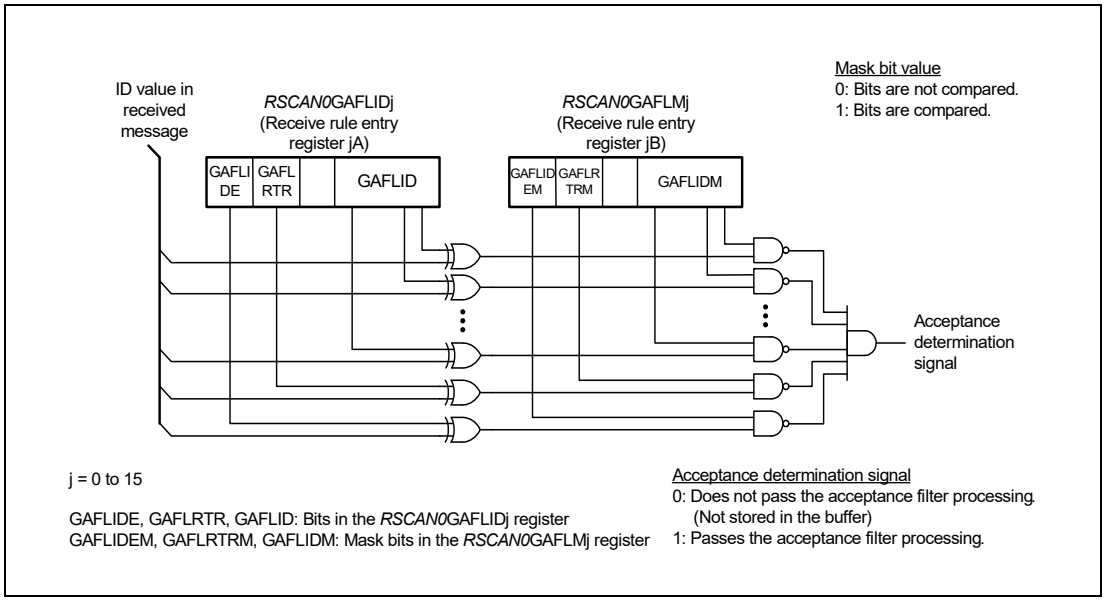


Figure 16.7 Acceptance Filter Function

16.4.3.3 DLC Filter Processing

When the DCE bit in the RSCAN0GCFG register is set to 1 (DLC check is enabled), DLC filter processing is added to messages that passed through the acceptance filter processing. When the DLC value in a message is equal to or larger than the DLC value set in the receive rule, the message passes through the DLC filter processing.

When a message has passed through the DLC filter processing with the DRE bit in the RSCAN0GCFG register set to 0 (DLC replacement is disabled), the DLC value of the received message is stored in the buffer. In this case, all the data bytes in the received message are stored in the buffer.

When a message has passed through the DLC filter processing with the DRE bit in the RSCAN0GCFG register set to 1 (DLC replacement is enabled), the DLC value of the receive rule is stored in the buffer instead of the DLC value of the received message. In this case, a value of 00_H is written to data bytes that are larger than the DLC value of the receive rule.

When the DLC value of the received message is smaller than that of the receive rule, the message does not pass through the DLC filter processing. In this case, the message is not stored in the receive buffer or the FIFO buffer and the DEF flag in the RSCAN0GERFL register is set to 1 (a DLC error is present).

16.4.3.4 Routing Processing

Messages that passed through the acceptance filter processing and the DLC filter processing are stored in receive buffers, receive FIFO buffers, or transmit/receive FIFO buffers set to receive mode or gateway mode. Message storage destination is set by the GAFLRMV and GAFLRMDP[6:0] bits in the RSCAN0GAFLP0j register (j = 0 to 15) and by the RSCAN0GAFLP1j register. Messages that passed through the acceptance filter processing and the DLC filter processing can be stored in up to 8 buffers.

16.4.3.5 Label Addition Processing

It is possible to add 12-bit label information to messages that passed through the filter processing and store them in buffers. This label information is set in the GAFLPTR[11:0] bits of the RSCAN0GAFLP0j register.

16.4.3.6 Mirror Function Processing

The mirror function allows reception of messages transmitted from the own CAN node. The mirror function is made available by setting the MME bit in the RSCAN0GCFG register to 1 (mirror function is enabled).

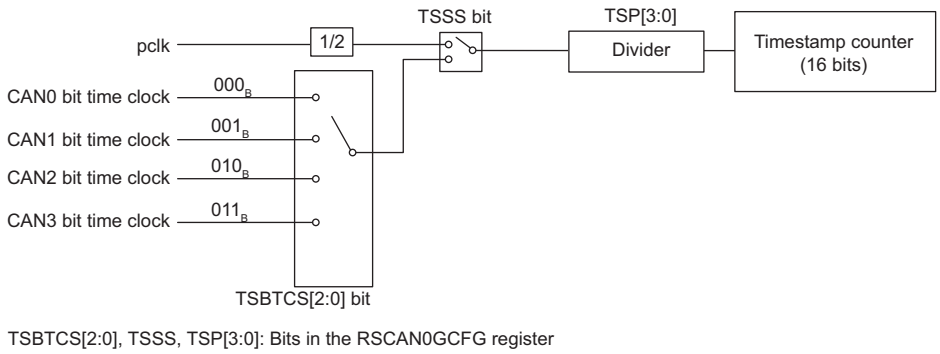
When the mirror function is in use, receive rules for which the GAFLLB bit in the RSCAN0GAFLIDj register is set to 0 are applied to the data processing for messages received from other CAN nodes. When messages transmitted from the own CAN node are received, receive rules for which the GAFLLB bit is set to 1 are used for data processing.

16.4.3.7 Timestamp

The timestamp counter is a 16-bit free-running counter used for recording the message receive time. The timestamp counter value is fetched at the timing of the start-of-frame (SOF) of a message and is then stored in a receive buffer or a FIFO buffer together with the message ID and data. pclk/2 or CANm bit time clock (m = 0 to 3) is selectable as a timestamp counter clock source by the TSBTCS[2:0] bits in the RSCAN0GCFG register. The clock obtained by dividing the selected clock source by the TSP[3:0] value of the RSCAN0GCFG register is used as the count source of the timestamp counter.

When the CANm bit time clock is used as a clock source, the timestamp counter stops when the corresponding channel transitions to channel reset mode or channel halt mode. When pclk/2 is used as a clock source, the timestamp function is not affected by channel mode.

The timestamp counter value is reset to 0000_H by setting the TSRST bit in the RSCAN0GCTR register to 1.



TSBTCS[2:0], TSSS, TSP[3:0]: Bits in the RSCAN0GCFG register

Note: When specifying pclk as the timestamp counter count source, set bits TSBTCS[2:0] to 000_B.

Figure 16.8 Timestamp Function Block Diagram

16.4.4 Transmission Functions

There are three types of transmission.

- Transmission using transmit buffers:
  Each channel has 16 buffers.
- Transmission using transmit/receive FIFO buffers (transmit mode):
  Each channel has three FIFO buffers. Up to 128 messages can be saved in a single FIFO buffer. Each FIFO buffer is used with a link to a transmit buffer. Only the message to be transmitted next in a FIFO buffer becomes a target of transmit priority determination. Messages are transmitted sequentially on a first-in, first-out basis.
- Transmission using transmit queues:
  Up to 16 transmit buffers per channel can be allocated to transmit queues. Transmit buffer number (16 x m) + 15 is used as a common access window. Transmit buffers are allocated to transmit queues in descending order of buffer numbers. All messages in transmit queues, which are targets of priority determination, are transmitted in the order of ID numbers.

Figure 16.9 shows the allocation of transmit queues and transmit/receive FIFO buffer link.

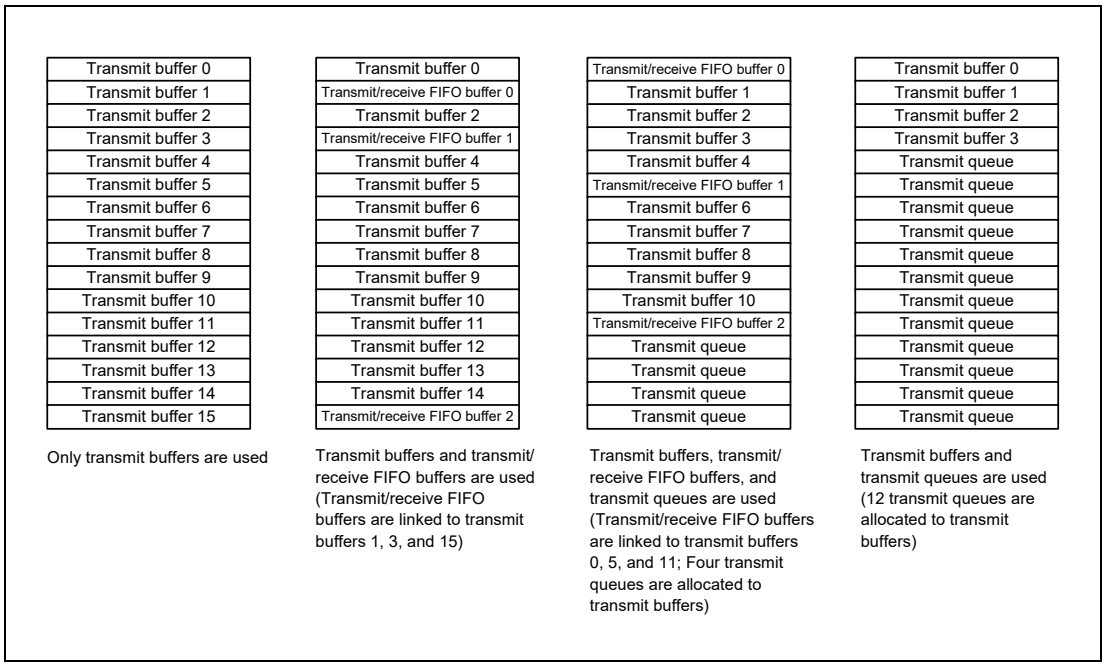


Figure 16.9 Allocation of Transmit Queues and Transmit/Receive FIFO Buffer Links

16.4.4.1 Transmit Priority Determination

If transmit requests are issued from multiple buffers or queues in the same channel, transmit priority is determined.

The priority is determined by using one of the following methods.

- ID priority (TPRI bit = 0)
- Transmit buffer number priority (TPRI bit = 1)

The setting of the TPRI bit in the RSCAN0GCFG register is effective in all CAN channels.

When the TPRI bit is set to 0, messages are transmitted according to the priority of stored message IDs. Priority of IDs conforms to the CAN bus arbitration specification defined in the CAN specification. Targets of priority determination are IDs of messages placed in transmit buffers, transmit/receive FIFO buffers (set to transmit mode or gateway mode), and transmit queues. When one or more transmit queues are used, select the ID priority method. When transmit/receive FIFO buffers are used, the oldest message in a FIFO buffer becomes a target of priority determination. When a message is being transmitted from a transmit/receive FIFO buffer, the next message in the same FIFO buffer becomes a target of priority determination. When transmit queues are used, all messages in transmit queues are targets of priority determination. If the same ID is set for two or more buffers, the buffer with a lower number takes precedence.

When the TPRI bit is set to 1, the message in the transmit buffer whose number is the lowest among buffers having transmit requests is transmitted first. When transmit/receive FIFO buffers are linked to transmit buffers, transmit priority is determined according to the linked transmit buffer numbers.

When messages are retransmitted due to an arbitration lost or an error, transmit priority determination is made again regardless of the TPRI bit.

16.4.4.2 Transmission Using Transmit Buffers

Setting the transmit request bit (TMTR bit in the RSCAN0TMCp register) of a transmit buffer to 1 (transmission is requested) allows transmission of data frames or remote frames.

Transmit result is shown by the TMTRF[1:0] flag in the corresponding RSCAN0TMSTSp register (p = 0 to 63). When transmit completes successfully, the TMTRF[1:0] flag is set to 10_B (transmission has been completed (without transmit abort request)) or 11_B (transmission has been completed (with transmit abort request)).

16.4.4.3 Transmit Abort Function

Setting the TMTAR bit in the RSCAN0TMCp register to 1 (transmit abort is requested) cancels the transmit request from a transmit buffer for which the TMTRM bit in the RSCAN0TMSTSp register is set to 1 (a transmit request is present). When transmit abort is completed, the TMTRF[1:0] flag of the RSCAN0TMSTSp register is set to 01_B (transmit abort has been completed) and the transmit request is canceled (clearing the TMTRM bit to 0).

A message that is being transmitted or a message to be transmitted next according to the transmit priority determination cannot be aborted. However, if an arbitration lost or an error has occurred while a message for which the TMTAR bit is set to 1 is being transmitted, retransmission is not performed.

16.4.4.4 One-Shot Transmission Function (Retransmission Disabling Function)

When the TMOM bit of the RSCAN0TMCp register is set to 1 (one-shot transmission is enabled), transmission is performed only once. Even if an arbitration lost or an error occurs, retransmission is not performed.

One-shot transmission result is shown by the TMTRF[1:0] flag in the corresponding RSCAN0TMSTSp register. When one-shot transmission completes successfully, the TMTRF[1:0] flag is set to 10_B or 11_B. When an arbitration lost or an error has occurred, the TMTRF[1:0] flag is set to 01_B (transmit abort has been completed).

16.4.4.5 Transmission Using FIFO Buffers

A single transmit/receive FIFO buffer can save as many messages as specified by the CFDC[2:0] bits of the RSCAN0CFCCk register (k = 0 to 11). Messages are transmitted sequentially on a first-in, first out basis.

Transmit/receive FIFO buffers are linked to transmit buffers selected by the CFTML[3:0] bits in the RSCAN0CFCCk register. When the CFE bit of the RSCAN0CFCCk register is set to 1 (transmit/receive FIFO buffers are used), transmit/receive FIFO buffers become targets of transmit priority determination. Priority determination is applied only to the message to be transmitted next in a FIFO buffer.

When the CFE bit is set to 0 (no transmit/receive FIFO buffer is used), the CFEMP flag is set to 1 (the transmit/receive FIFO buffer contains no message (buffer empty)) at the timing below.

- The transmit/receive FIFO buffer becomes empty immediately when the message in it is not being transmitted or is not to be transmitted next.
- The transmit/receive FIFO buffer becomes empty after transmission completion, CAN bus error detection, or arbitration lost when the message in it is being transmitted or to be transmitted next.

When the CFE bit is cleared to 0, all messages in transmit/receive FIFO buffers are lost and messages cannot be stored in FIFO buffers. Confirm that the CFEMP flag is set to 1 before setting the CFE bit to 1 again.

16.4.4.6 Interval Transmission Function

To transmit messages continuously from the same transmit/receive FIFO buffer that is set to transmit mode or gateway mode, message transmission interval time can be set.

Immediately after the first message has been transmitted successfully from the FIFO buffer with the CFE bit set to 1 in the RSCAN0RSCAN0CFCCk register, the interval timer starts counting (after EOF7 of the CAN protocol). After that, when the interval time has passed, the next message is transmitted. The interval timer stops in channel reset mode or by clearing the CFE bit to 0.

The interval time is set by the CFITT[7:0] bits in the RSCAN0CFCCk register. When the interval timer is not used, set the CFITT[7:0] bits to 00_H.

Select an interval timer count source by the CFITR and CFITSS bits in the RSCAN0CFCCk register. When the CFITR and CFITSS bits are set to 00_B, the clock obtained by dividing pclk/2 by the ITRCP[15:0] value is used as a count source. When the CFITR and CFITSS bits are set to 10_B, 10 clocks obtained by dividing pclk/2 by the ITRCP[15:0] value are used as a count source. When the CFITR and CFITSS bits are set to $x1_B$, the CANm bit time clock is used as a count source.

The interval time is calculated by the following formulae where m is the ITRCP[15:0] value and n is the CFITT[7:0] value.

- When CFITR and CFITSS bits are 00_B (fPBA is the frequency of pclk):

$$\frac{1}{fPBA} \times 2 \times m \times n$$

- When CFITR and CFITSS bits are 10_B:

$$\frac{1}{fPBA} \times 2 \times m \times 10 \times n$$

- When CFITR and CFITSS bits are $x1_B$ (fCANBIT is the frequency of CANm bit time clock):

$$\frac{1}{fCANBIT} \times n$$

Figure 16.10 shows the interval timer block diagram.

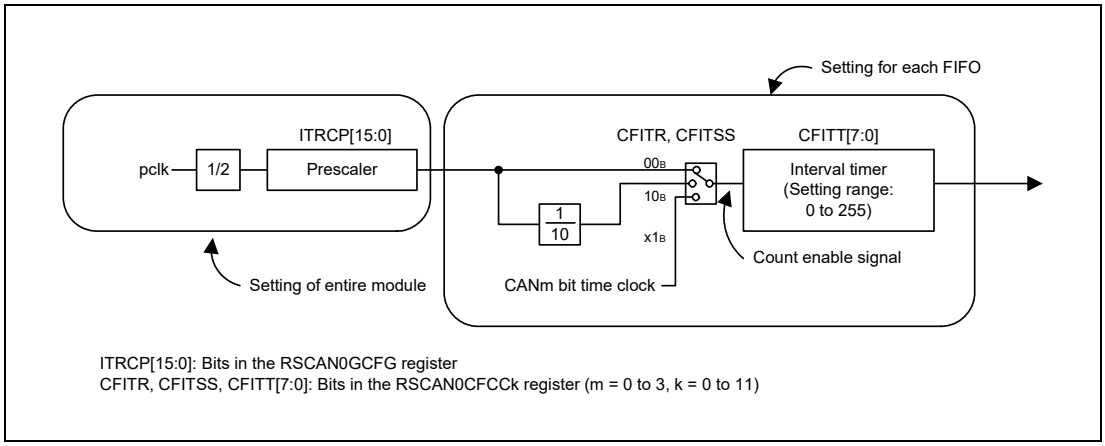


Figure 16.10 Interval Timer Block Diagram

Figure 16.11 shows the interval timer timing chart.

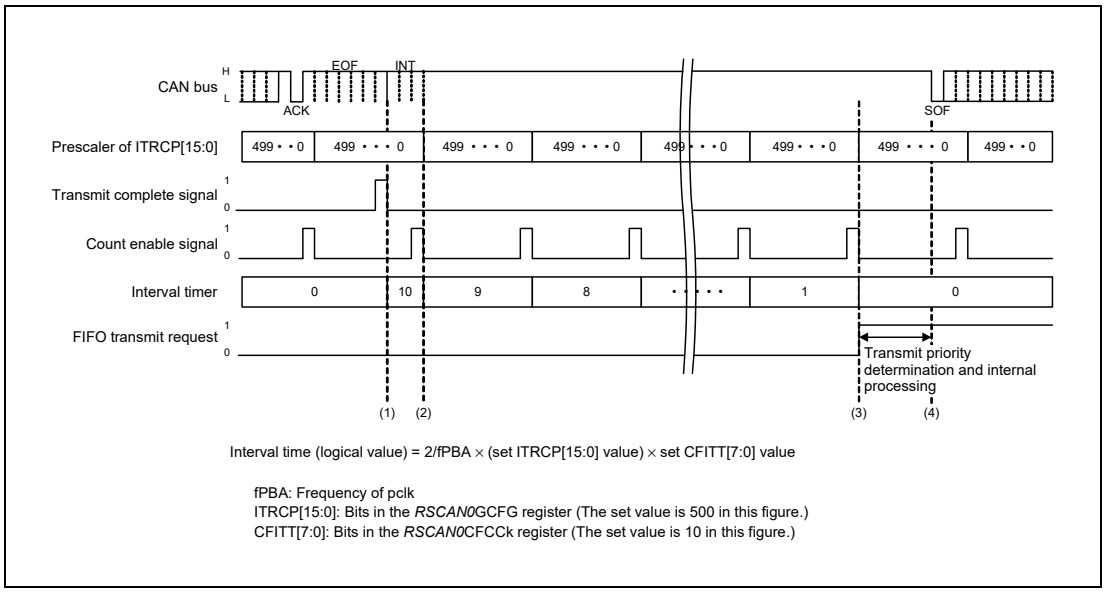


Figure 16.11 Interval Timer Timing Chart

1. The interval timer starts counting upon completion of transmission. Since the prescaler is not initialized at the time of transmission completion, the first interval time has a margin of error of up to one interval timer count.
2. The interval timer is decremented by 1 by the next count enable signal.
3. When the interval timer has decreased to 0, the transmit/receive FIFO buffer issues a transmit request.
4. When the transmit/receive FIFO buffer is determined for the next transmission by the priority determination, it starts transmitting data. Transmission starts usually with a delay of three CANm bit time clock cycles or less from the issue of transmit request.
   Transmission start may delay up to 504 pclk clock cycles if several internal processing including receive filter processing, message routing, and transmit priority determination are performed at all channels.

16.4.4.7 Transmission Using Transmit Queues

Three to sixteen buffers are allocated to a transmit queue for each channel, and the transmit buffer (16 × m) + 15 is used as a common access window.

All messages in a transmit queue are targets of transmit priority determination and are transmitted in the ID priority order regardless of the storing order. If two messages having the same ID are stored in a transmit queue, these messages are not always be transmitted in the storing order in the transmit queue.

Setting the TXQE bit in the RSCAN0TXQCCm register to 0 disables transmit queues. When the TXQE bit is set to 0, the TXQEMP flag in the RSCAN0TXQSTSm register is set to 1 (the transmit queue contains no message (transmit queue empty)) at the timing below.

- The transmit queue becomes empty immediately when the message in it is not being transmitted or is not to be transmitted next.
- The transmit queue becomes empty after transmission completion, CAN bus error detection, or arbitration lost when the message in it is being transmitted or to be transmitted next.

When the TXQE bit is cleared to 0, all messages in transmit queues are lost and messages cannot be stored in transmit queues. Confirm that the TXQEMP flag is set to 1 before setting the TXQE bit to 1 again.

16.4.4.8 Transmit History Function

Information of transmitted messages can be stored in the transmit history buffer. Each channel has a single transmit history buffer that can contain 16 sets of transmit history data.

A message transmit source buffer type can be selected by the THLDTE bit in the RSCAN0THLCCm register. Whether to store transmit history data for each message can be set by the THLEN bit in the RSCAN0CFIDk register (k = 0 to 11).

Storing the transmit history data may be delayed by up to 144 pclk clock cycles after successful completion of transmission.

- Buffer type 001_B: Transmit buffer
  010_B: Transmit/receive FIFO buffer
  100_B: Transmit queue
- Buffer number Number assigned to source transmit buffer, transmit queue, or transmit/receive FIFO buffer. This number depends on buffer types. See **Table 16.91**.
- Label data Label information of transmitted message

Table 16.91 Transmit History Data Buffer Numbers

Buffer type / Buffer No.	001_B	010_B	100_B
0000_B	Transmit buffer 16 × m + 0	Numbers of transmit buffers linked to transmit/receive FIFO buffers by the CFTML[3:0] bits in the RSCAN0CFCCk register (k = 0 to 11)	Numbers of transmit buffers allocated to the transmit queue that performed transmission
0001_B	Transmit buffer 16 × m + 1		
0010_B	Transmit buffer 16 × m + 2		
0011_B	Transmit buffer 16 × m + 3		
0100_B	Transmit buffer 16 × m + 4		
0101_B	Transmit buffer 16 × m + 5		
0110_B	Transmit buffer 16 × m + 6		
0111_B	Transmit buffer 16 × m + 7		
1000_B	Transmit buffer 16 × m + 8		
1001_B	Transmit buffer 16 × m + 9		
1010_B	Transmit buffer 16 × m + 10		
1011_B	Transmit buffer 16 × m + 11		
1100_B	Transmit buffer 16 × m + 12		
1101_B	Transmit buffer 16 × m + 13		
1110_B	Transmit buffer 16 × m + 14		
1111_B	Transmit buffer 16 × m + 15		

Label data is used to identify each message. A unique label data can be added to each message transmitted from a transmit buffer, transmit queue, or transmit/receive FIFO buffer.

Transmit history data can be read from the RSCAN0THLACCm register. If it is attempted to store new transmit history data while the buffer is full, the buffer overflows and the new data is discarded.

16.4.5 Gateway Function

When a transmit/receive FIFO buffer is set to gateway mode, received massages can be transmitted from an arbitrary channel without CPU intervention.

When a transmit/receive FIFO buffer for which the CFM[1:0] bits of the RSCAN0CFCCk register are set to 10_B (gateway mode) is selected by the RSCAN0GAFLP1j register, messages that passed through the filter processing of the receive rule are stored in the specified transmit/receive FIFO buffer and are automatically transmitted from the buffer.

Messages stored in a transmit/receive FIFO buffer are transmitted sequentially on a first-in, first-out basis. Only the message to be transmitted next becomes a target of transmit priority determination.

Transmit/receive FIFO buffers in the gateway mode are disabled by setting the CFE bit in the RSCAN0CFCCk register to 0. When the CFE bit is set to 0, the CFEMP flag becomes 1 at the timing below.

- The transmit/receive FIFO buffer becomes empty immediately when the message in it is not being transmitted or is not to be transmitted next.
- The transmit/receive FIFO buffer becomes empty after transmission completion, CAN bus error detection, or arbitration lost when the message in it is being transmitted or to be transmitted next.

When the CFE bit is cleared to 0, all messages in transmit/receive FIFO buffers are lost and messages can no longer be stored in transmit/receive FIFO buffers. Confirm that the CFEMP flag is set to 1 before setting the CFE bit to 1 again.

16.4.6 Test Function

The test function is classified into communication tests and global tests.

- Communication tests: Performed for each channel.
  - Standard test mode
  - Listen-only mode
  - Self-test mode 0 (external loopback mode)
  - Self-test mode 1 (internal loopback mode)
- Global tests: Performed in entire module
  - RAM test (read/write test)
  - Inter-channel communication test

16.4.6.1 Standard Test Mode

Standard test mode allows CRC test.

16.4.6.2 Listen-Only Mode

Listen-only mode allows reception of data frames and remote frames. Only recessive bits are transmitted on the CAN bus, and the ACK bit, overload flag, and active error flag are not transmitted.

Listen-only mode is available for detecting the communication speed.

Do not make a transmit request from any buffer or queue in listen-only mode.

Figure 16.12 shows the connection when listen-only mode is selected.

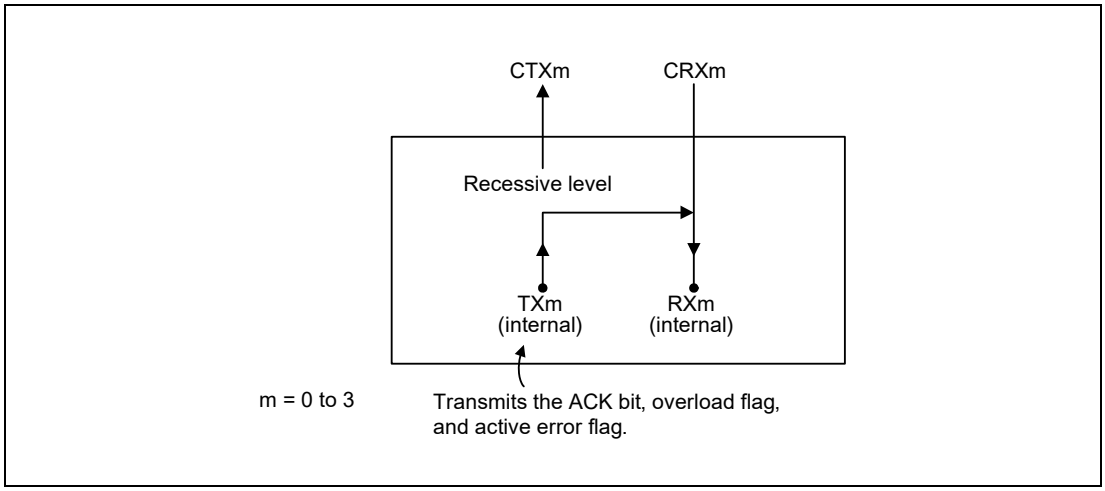


Figure 16.12 Connection When Listen-Only Mode is Selected

16.4.6.3 Self-Test Mode (Loopback Mode)

In self-test mode, messages transmitted from a channel are compared with the receive rule of the same channel. Messages are stored in a buffer if they have passed through the filter processing. Messages transmitted from other CAN nodes are compared only with the receive rule for which the GAFLLB bit in the RSCAN0GAFLIDj register (j = 0 to 15) is set to 0 (when a message transmitted from another CAN node is received).

If the mirror function and self-test mode are both enabled, the self-test mode setting takes precedence.

16.4.6.4 Self-Test Mode 0 (External Loopback Mode)

Self-test mode 0 is used to perform a loopback test within a channel including the CAN transceiver.

In self-test mode 0, transmitted messages are handled as messages received through the CAN transceiver and are stored in a buffer. An ACK bit is generated to receive messages transmitted from the own CAN node.

Figure 16.13 shows the connection when self-test mode 0 is selected.

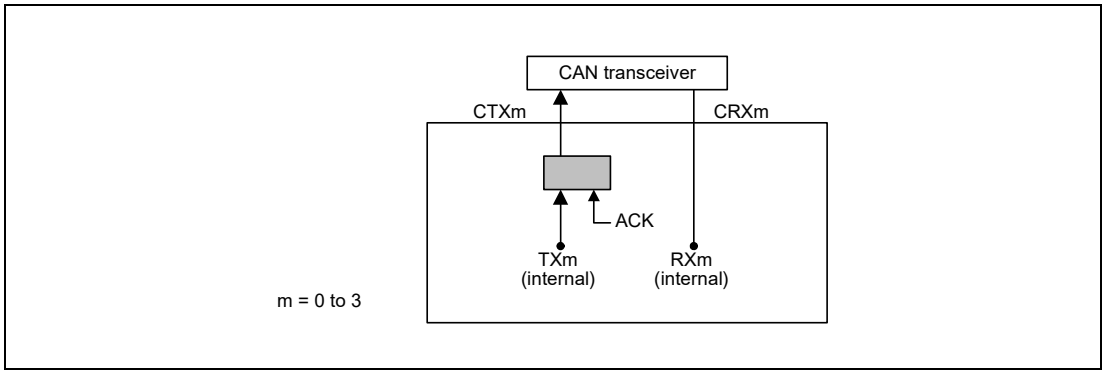


Figure 16.13 Connection When Self-Test Mode 0 is Selected

16.4.6.5 Self-Test Mode 1 (Internal Loopback Mode)

In self-test mode 1, transmitted messages are handled as received messages and are stored in a buffer. An ACK bit is generated to receive messages transmitted from the own CAN node.

In self-test mode 1, internal feedback from the internal TXm pin (m = 0 to 3) to the internal RXm pin is performed. The external CRXm pin input is isolated. The external CTXm pin outputs only recessive bits.

Figure 16.14 shows the connection when self-test mode 1 is selected.

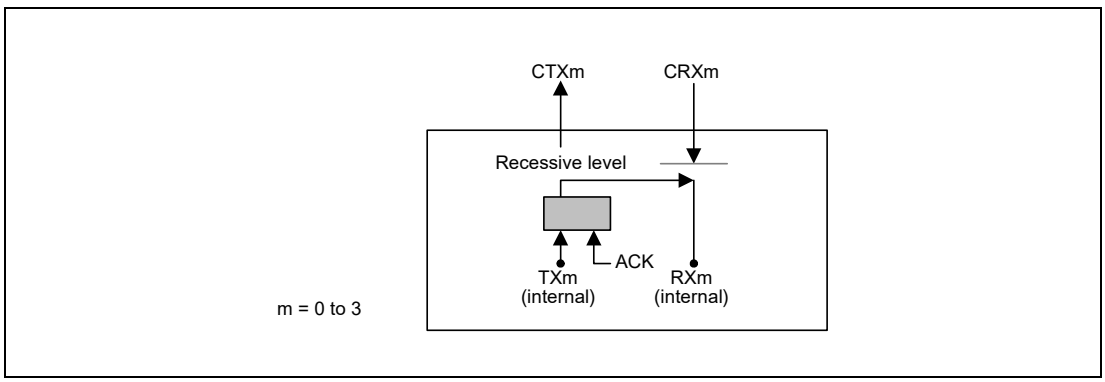


Figure 16.14 Connection When Self-Test Mode 1 is Selected

16.4.6.6 RAM Test

The RAM test function allows accesses to all CAN RAM addresses. RAM initialization which is performed after resetting the MCU does not initialize all CAN RAM areas.

When the RAM test function is used, the RAM is divided into pages of 256 bytes each. A RAM test page is selected by the RTMPS[6:0] bits in the RSCAN0GTSTCFG register. Data in the set page can be read from and written to the RSCAN0RPGACCr register (r = 0 to 63). The available total RAM size is 12160 bytes ($2F80_H$).

16.4.6.7 Inter-Channel Communication Test

The inter-channel communication test function allows communication test by internally and mutually connecting CAN channels. During this test, channels are isolated from the external CAN bus.

Before starting data transmission/reception in channel communication mode, make transmission/reception settings for each channel. Channels not included in the test must be placed in channel halt mode.

Figure 16.15 shows the connection for inter-channel communication test.

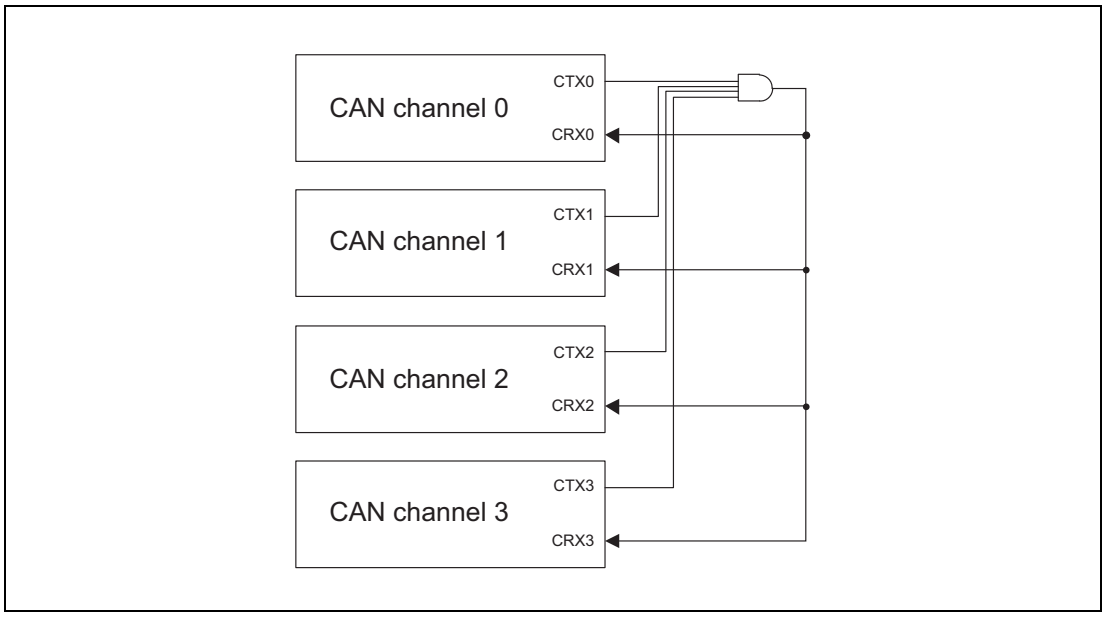
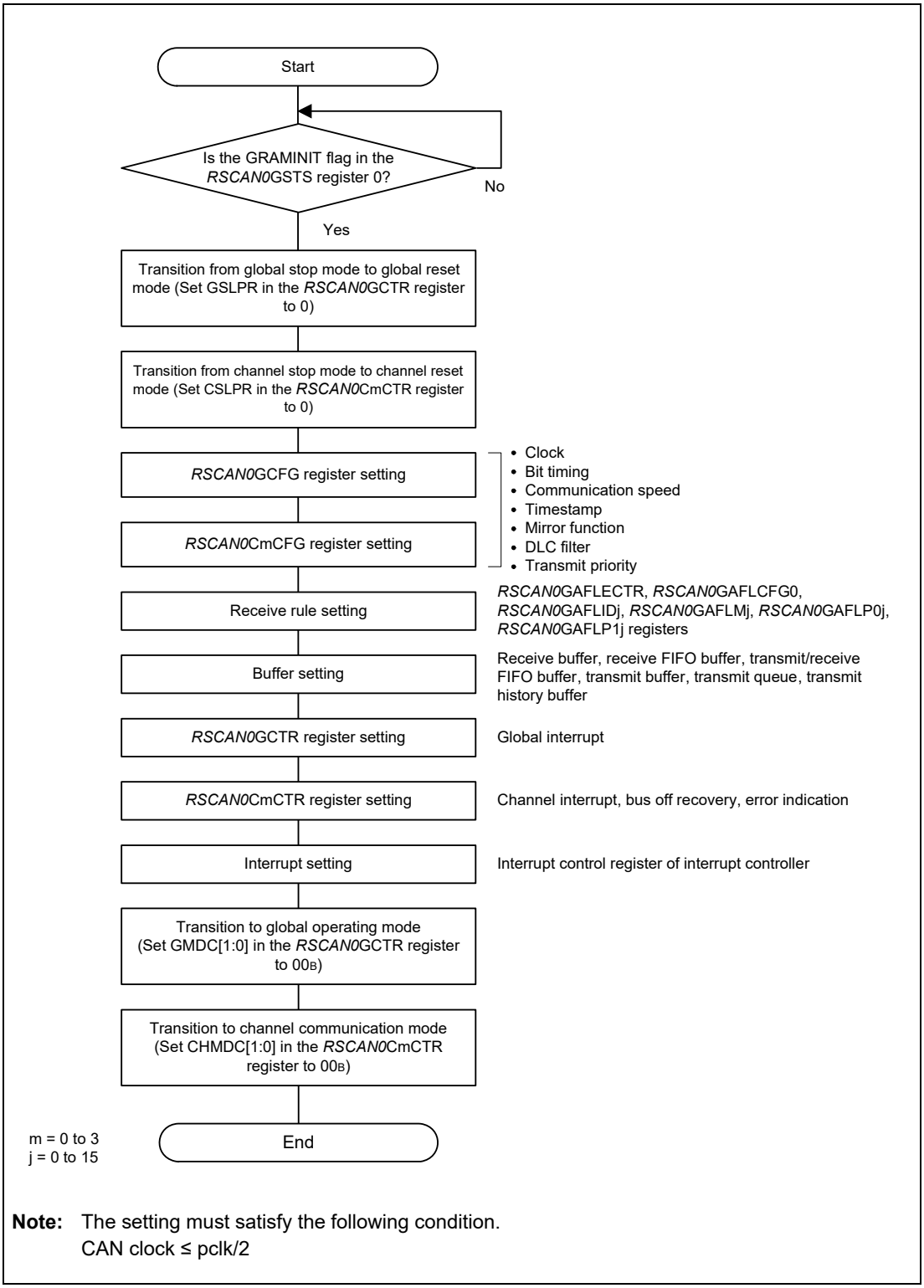


Figure 16.15 Connection for Inter-Channel Communication Test

16.5 Procedures

16.5.1 Initial Settings

The RS-CAN module initializes the CAN RAM after the MCU is reset. The RAM initialization time is 6082 cycles/pclk. The GRAMINIT flag in the RSCAN0GSTS register is set to 1 (CAN RAM initialization is ongoing) during the RAM initialization and is cleared to 0 (CAN RAM initialization is finished) when the initialization is completed. Configure CAN settings after the GRAMINIT flag is cleared to 0. **Figure 16.16** shows the CAN setting procedure after the MCU is reset.



Note: The setting must satisfy the following condition.
CAN clock ≤ pclk/2

Figure 16.16 CAN Setting Procedure after the MCU is Reset

16.5.1.1 Clock Setting

Set the CAN clock (fCAN) which is the clock source of the RS-CAN module. Select clkc or clk_xincan by the DCS bit in the RSCAN0GCFG register.

16.5.1.2 Bit Timing Setting

In the CAN protocol, one bit of a communication frame consists of three segments: SS, TSEG1, and TSEG2. Of these, the two segments, TSEG1 and TSEG2 can be set by the RSCAN0CmCFG register for each channel. Sample point timing can be determined by setting these two segments. This timing can be adjusted in units of 1 Time Quantum (referred to as Tq hereinafter). 1 Tq equals to one CANmTq clock cycle. The CANmTq clock is obtained by dividing the clock selected with the DCS bit of the RSCAN0GCFG register by the value of the BRP[9:0] bits of the RSCAN0CmCFG register.

Figure 16.17 shows the bit timing chart. **Table 16.92** shows an example of bit timing setting.

SS = 1 Tq fixed
Set TSEG1 to a range of 4 Tq to 16 Tq
Set TSEG2 to a range of 2 Tq to 8 Tq
Set SJW to a range of 1 Tq to 4 Tq
Set SS + TSEG1 + TSEG2 to a range of 8 Tq to 25 Tq
TSEG1 > TSEG2 ≥ SJW

- SS (synchronization segment):
  The SS is a segment that performs synchronization by monitoring the edge from recessive to dominant bits in the Interframe Space.
  Interframe Space consists of Intermission, Suspend Transmission, and Bus Idle. All nodes can start transmission during Bus Idle.
- TSEG1 (Time segment 1):
  TSEG1 is a segment that absorbs physical delay on the CAN network. The physical delay on the CAN network is twice of the total of the delay on the CAN bus, the delay in the input comparator, and the delay in the output driver.
- TSEG2 (Time segment 2):
  TSEG2 is a segment that compensates phase error due to an error in frequency.
- SJW (Resynchronization jump width)
  The SJW is a length to extend or reduce time segment to compensate an error in phase due to phase error.

Figure 16.17 Bit Timing Chart

Table 16.92 Example of Bit Timing Setting

1 Bit	Set Value (Tq)				Sampling Point (%) Note: See Figure 16.17.
	SS	TSEG1	TSEG2	SJW	
20Tq	1	12	7	1	65.00
	1	13	6	1	70.00

16.5.1.3 Communication Speed Setting

Set the CAN communication speed for each channel using the fCAN, baud rate prescaler division value (BRP[9:0] bits in the RSCAN0CmCFG register), and Tq count of one bit time.

Figure 16.18 shows the CAN clock control block diagram, and **Table 16.93** shows an example of the communication speed setting.

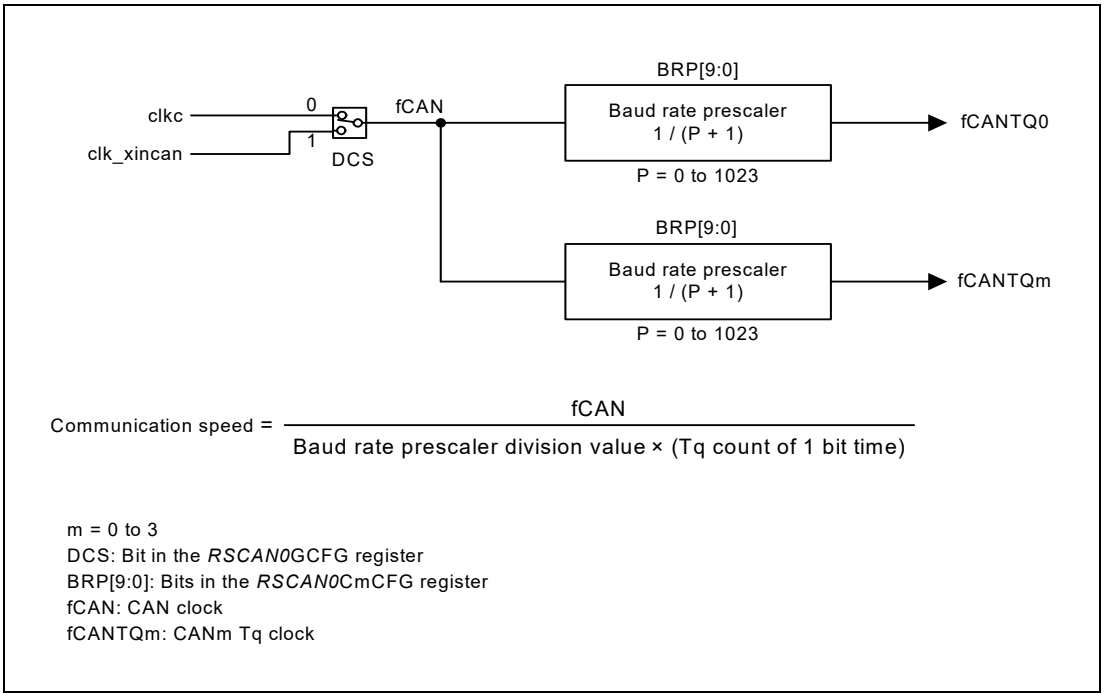


Figure 16.18 CAN Clock Control Block Diagram

Table 16.93 Example of Communication Speed Setting

fCAN / Communication speed	40 MHz
1 Mbps	8 Tq (5) 20 Tq (2)
500 Kbps	8 Tq (10) 20 Tq (4)
250 Kbps	8 Tq (20) 20 Tq (8)

Note: Values in ( ) are baud rate prescaler division values.

16.5.1.4 Receive Rule Setting

Receive rules can be set using receive rule-related registers.

Up to 16 receive rules can be registered per page. Specify pages 0 to 15 by the AFLPN[4:0] bits in the RSCAN0GAFLECTR register. Use the AFLDAE bit to enable or disable writing to the receive rule table.

Figure 16.19 shows the receive rule setting procedure.

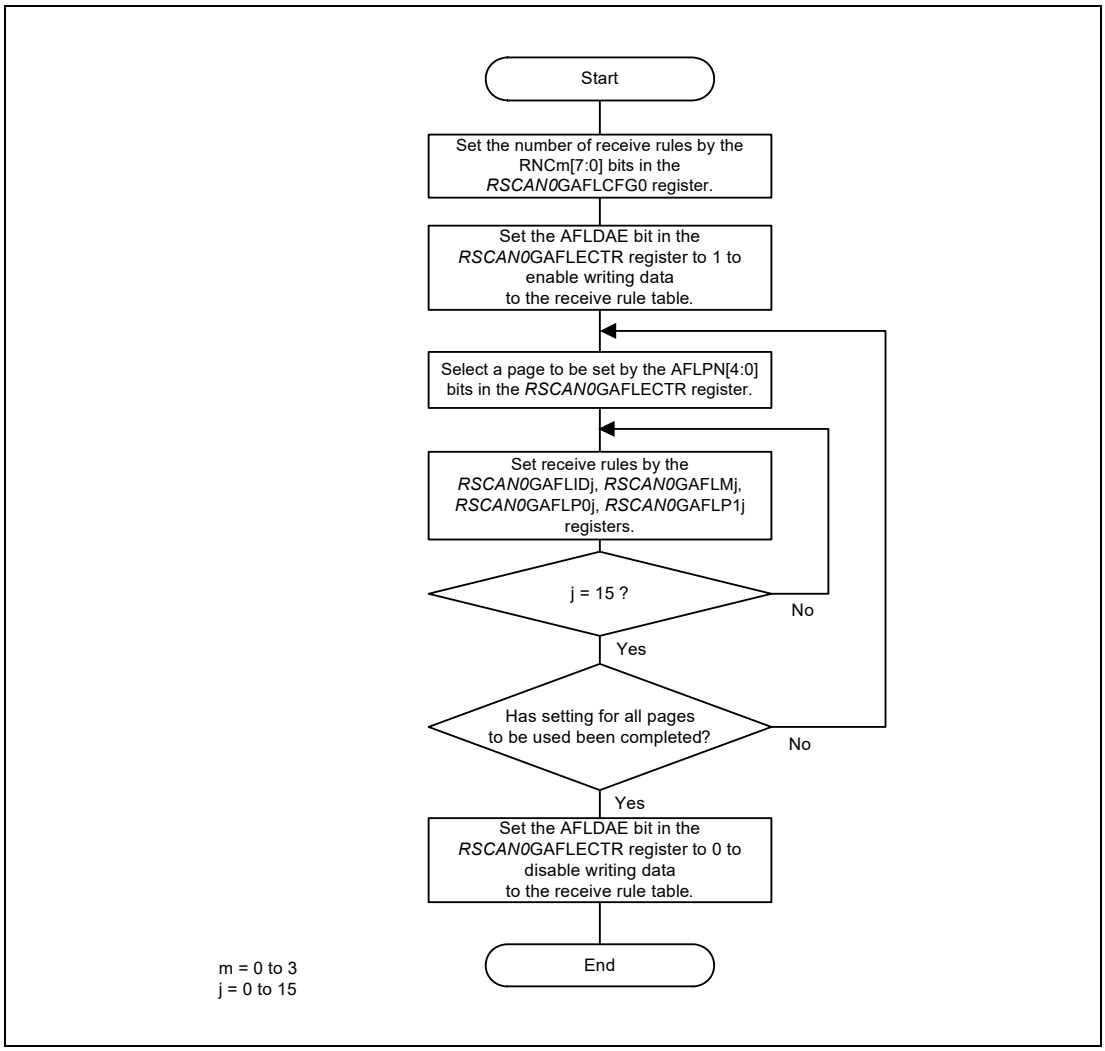
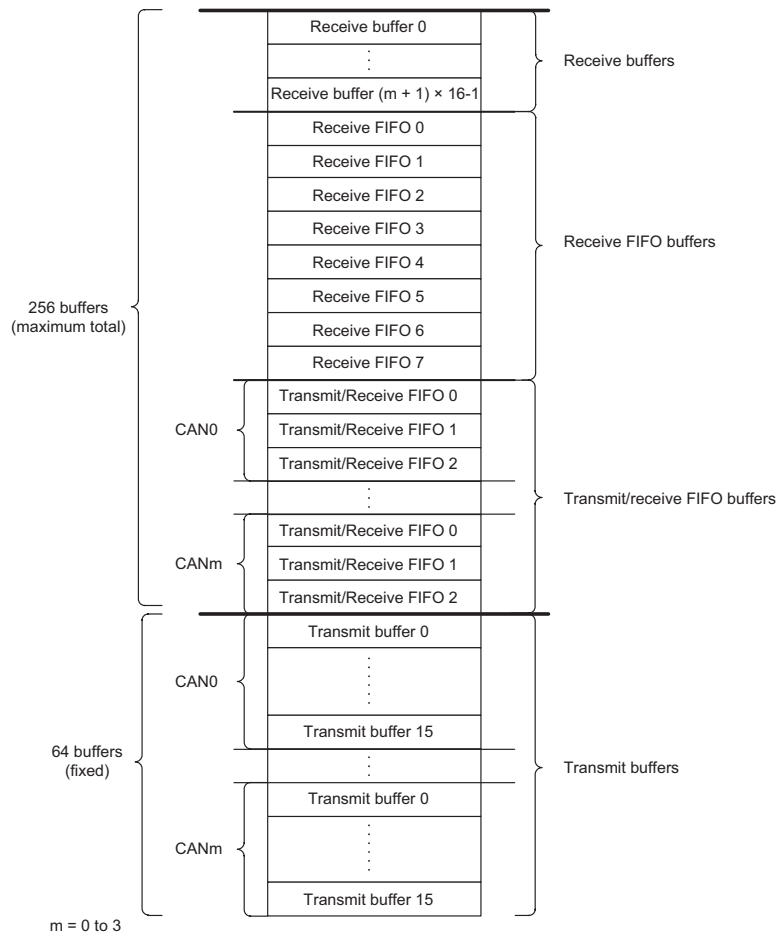


Figure 16.19 Receive Rule Setting Procedure

16.5.1.5 Buffer Setting

Set sizes and interrupt sources of buffers. For transmit/receive FIFO buffers that are set to transmit mode, set transmit buffers to be linked.

Figure 16.20 shows the buffer configuration. **Figure 16.21** shows the buffer setting procedure.



Note: Receive buffers, receive FIFO buffers, transmit/receive FIFO buffers, and transmit buffers are located in succession.

Figure 16.20 Buffer Configuration

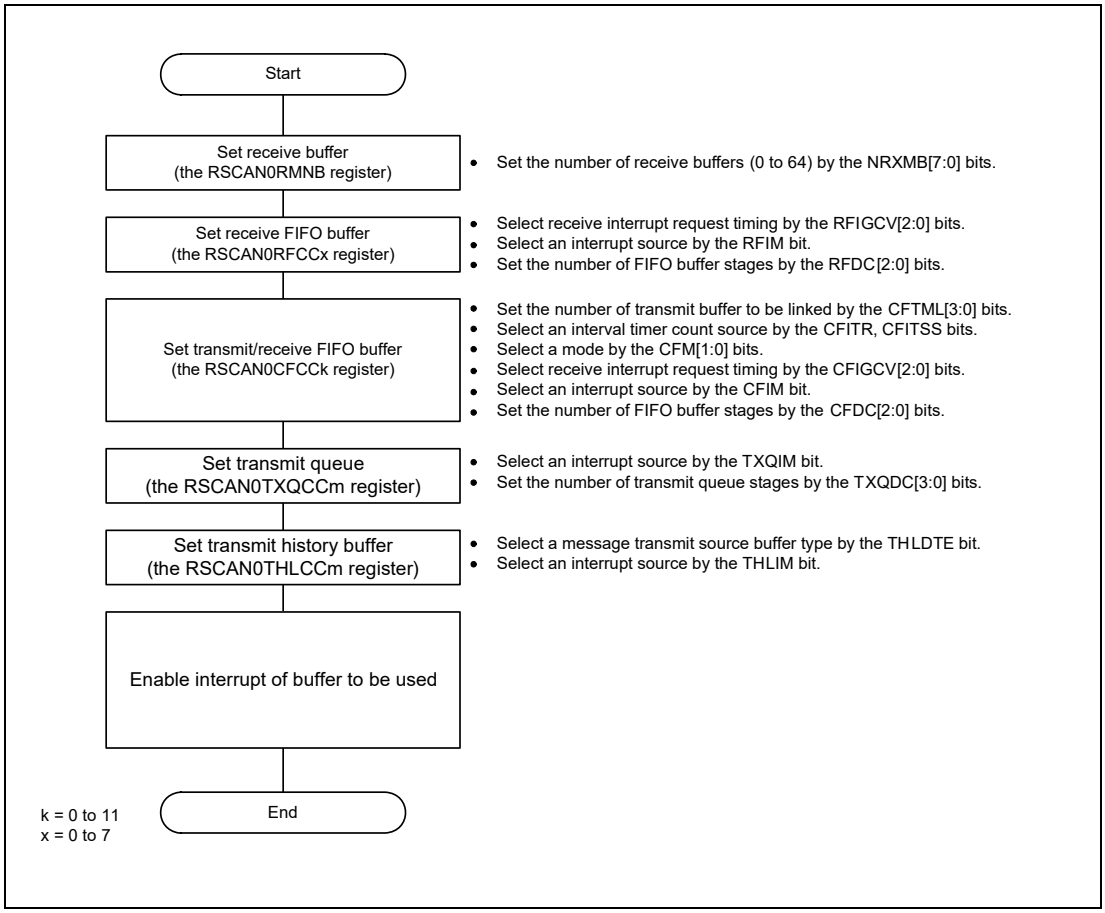


Figure 16.21 Buffer Setting Procedure

16.5.2 Procedure for Reception

16.5.2.1 Receive Buffer Reading Procedure

When the processing to store received messages in a receive buffer starts, the RMNSq flag in the RSCAN0RMNDy register (y = 0, 1, q = 0 to 63) is set to 1 (receive buffer q contains a new message). Messages can be read from the RSCAN0RMIDq, RSCAN0RMPTRq, RSCAN0RMDF0q, and RSCAN0RMDF1q registers. If the next message has been received before the current message is read from the receive buffer, the message is overwritten.

Figure 16.22 shows the receive buffer reading procedure.

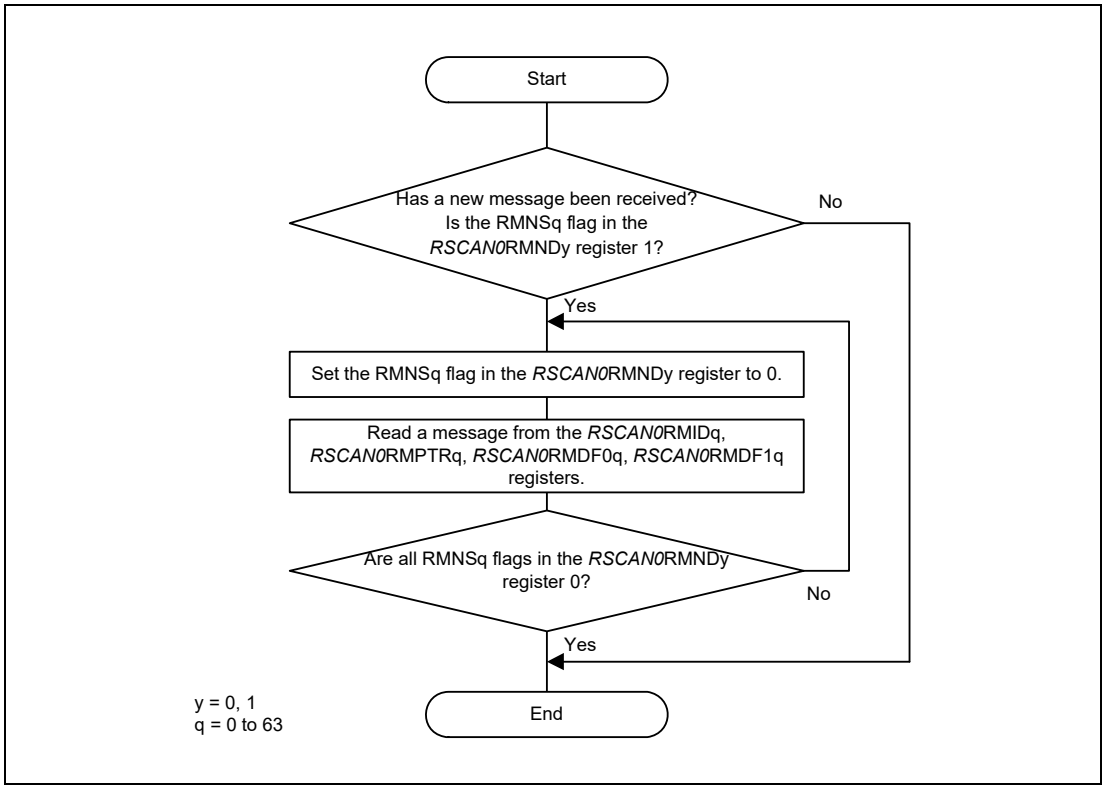


Figure 16.22 Receive Buffer Reading Procedure

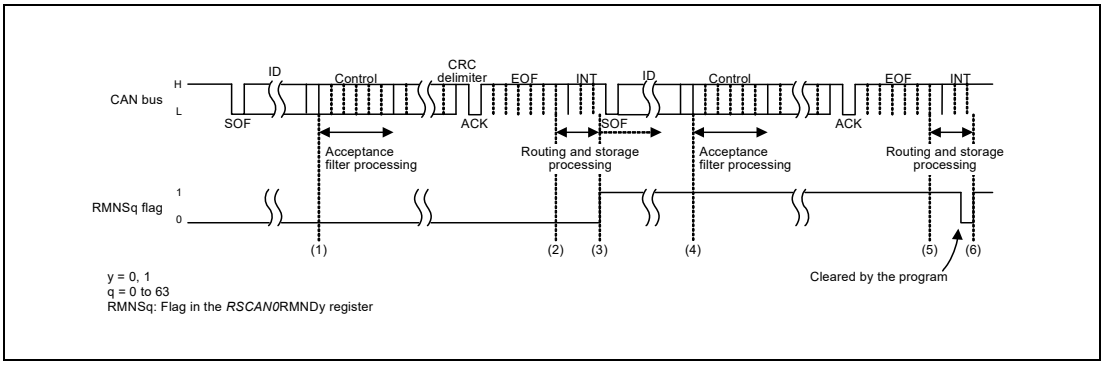


Figure 16.23 Receive Buffer Reception Timing Chart

(1) When the ID field of a message has been received, the acceptance filter processing starts.

(2) When the message matches the receive rule of the corresponding channel and has been successfully received, the routing processing to transfer the message to the specified buffer starts. When the DCE bit in the RSCAN0GCFG register is set to 1 (DLC check is enabled), the DLC filter processing starts at this time.

(3) When the message has passed through the DLC filter processing, the processing to store the message in the specified receive buffer starts.
When the message storage processing starts, the RMNSq flag in the corresponding RSCAN0RMNDy register is set to 1 (receive buffer n contains a new message). If any other channels are performing filter processing or transmit priority determination processing, the routing processing and the storage processing may be delayed.

(4) When the ID field of the next message has been received, the acceptance filter processing starts.

(5) When the message matches the receive rule of the corresponding channel and has been successfully received, the routing processing to transfer the message to the specified buffer starts. When the DCE bit in the RSCAN0GCFG register is set to 1 (DLC check is enabled), the DLC filter processing starts at this time.

(6) When the corresponding RMNSq flag is cleared to 0 (receive buffer n contains no new message), this flag is set to 1 again when the message storage processing starts. If the RMNSq flag remains 1, a new message is overwritten in the receive buffer. The RMNSq flag cannot be cleared to 0 during storage of messages.

16.5.2.2 FIFO Buffer Reading Procedure

When received messages have been stored in one or more receive FIFO buffers or transmit/receive FIFO buffers that are set to receive mode or gateway mode, the corresponding message count display counter (RFMC[7:0] bits in the RSCAN0RFSTSx register (x = 0 to 7) or CFMC[7:0] bits in the RSCAN0CFSTSk register (k = 0 to 11)) is incremented by 1. At this time, if the RFIE bit (receive FIFO interrupt is enabled) in the RSCAN0RFCCx register or the CFRXIE bit (transmit/receive FIFO reception interrupt is enabled) in the RSCAN0CFCCk register is set to 1, an interrupt request is generated. Received messages can be read from the RSCAN0RFIDx, RSCAN0RFPTRx, RSCAN0RFDF0x, and RSCAN0RFDF1x registers (receive FIFO buffers) or the RSCAN0CFIDk, RSCAN0CFPTRk, RSCAN0CFDF0k, and RSCAN0CFDF1k registers (transmit/receive FIFO buffers). Messages in FIFO buffers can be read sequentially on a first-in, first-out basis.

When the message count display counter value matches the FIFO buffer depth (a value set by the RFDC[2:0] bits in the RSCAN0RFCCx register or the CFDC[2:0] bits in the RSCAN0CFCCk register), the RFFLL or CFFLL flag is set to 1 (the receive FIFO buffer is full.).

When all messages have been read out of the FIFO buffer, the RFEMP flag in the RSCAN0RFSTSx register or the CFEMP flag in the RSCAN0CFSTSk register is set to 1 (the receive FIFO buffer contains no unread message (buffer empty)).

If the RFE bit or the CFE bit is cleared to 0 (no receive FIFO buffer is used) with the interrupt request flag (RFIF flag in the RSCAN0RFSTSx register or CFRXIF flag in the RSCAN0CFSTSk register) set to 1 (a receive FIFO interrupt request is present), the interrupt request flag is not automatically cleared to 0. Clear the interrupt request flag to 0 by the program.

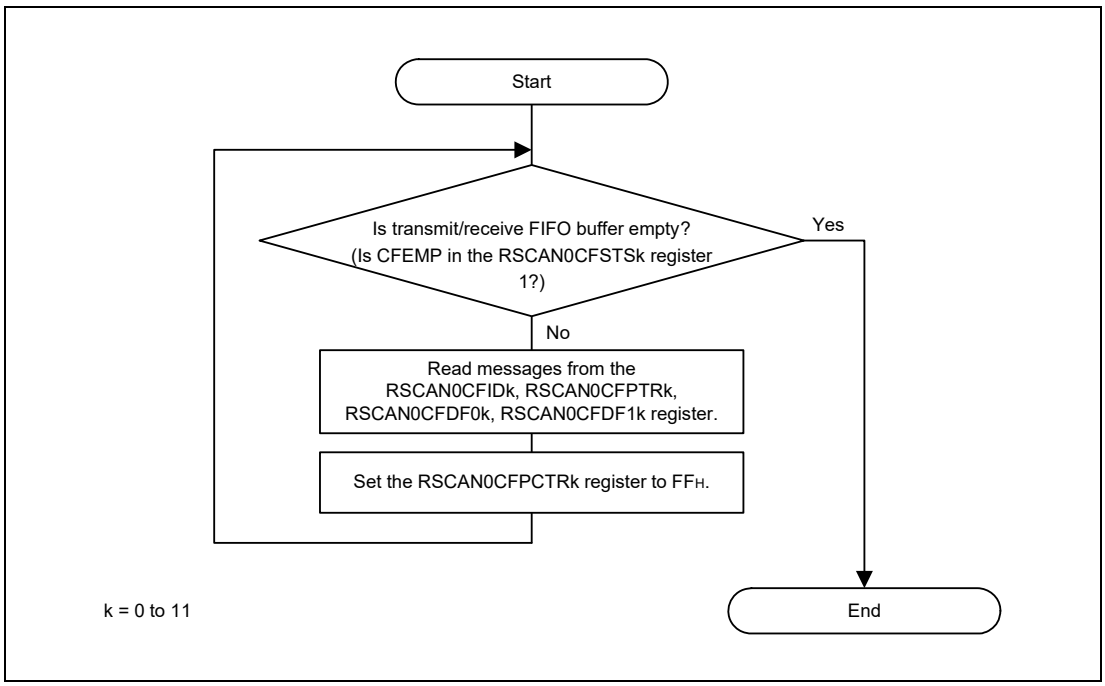


Figure 16.24 Transmit/Receive FIFO Buffer Reading Procedure

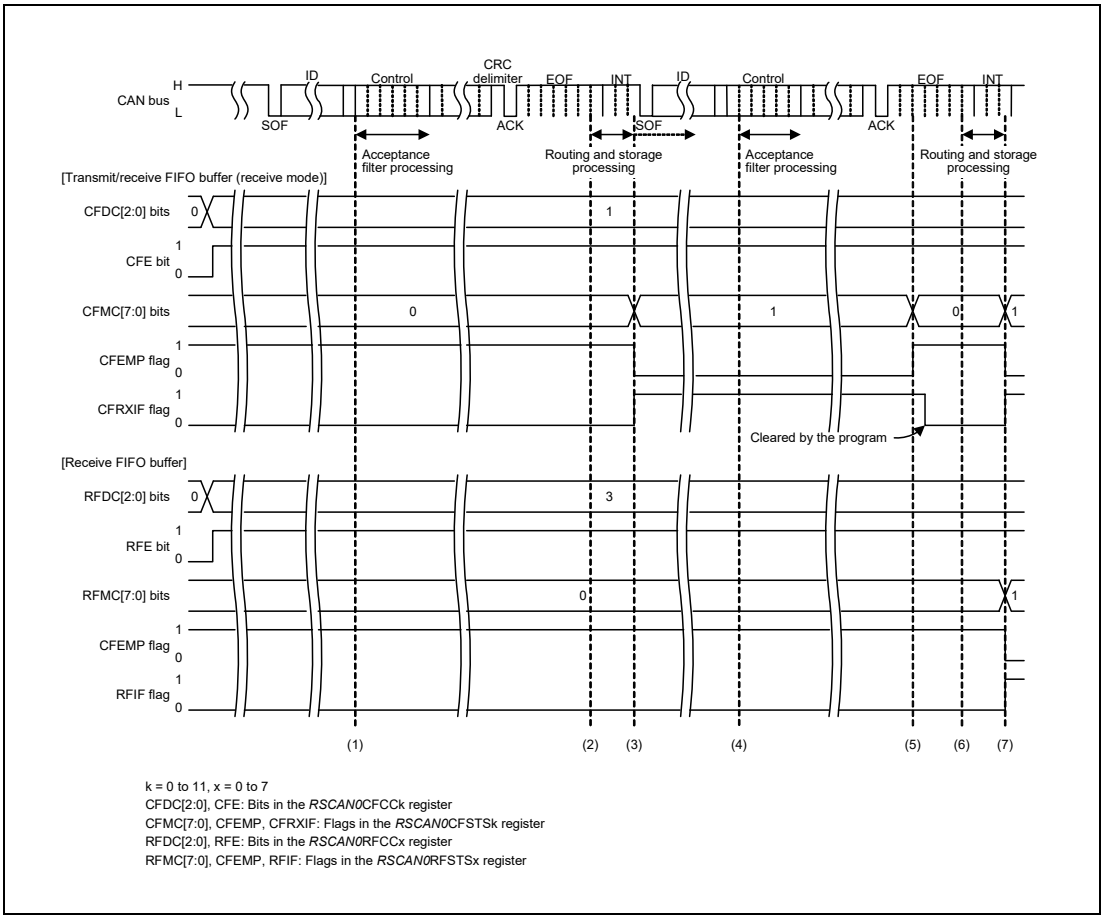


Figure 16.25 FIFO Buffer Receive Timing Chart

(1) When the ID field of a message has been received, the acceptance filter processing starts.

(2) When the message matches the receive rule of the corresponding channel and has been successfully received, the routing processing to transfer the message to the specified buffer starts. When the DCE bit in the RSCAN0GCFG register is set to 1 (DLC check is enabled), the DLC filter processing starts at this time.

(3) When the message has passed through the DLC filter processing and the CFE value in the RSCAN0CFCCk register is 1 (transmit/receive FIFO buffers are used) and the CFDC[2:0] value in the RSCAN0CFCCk register is 001_B or more, the message is stored in the transmit/receive FIFO buffer that is set to receive mode. The CFMC[7:0] value in the RSCAN0CFSTSk register is incremented by 1 and becomes 01_H. When the CFIM bit in the RSCAN0CFCCk register is set to 1 (a FIFO reception interrupt request is generated each time a message has been received), the CFRXIF flag in the RSCAN0CFSTSk register is set to 1 (a transmit/receive FIFO reception interrupt request is present). The CFRXIF flag can be cleared to 0 by the program.

(4) When the ID field of the next message has been received, the acceptance filter processing starts.

(5) Read received messages from the RSCAN0CFIDk, RSCAN0CFPTRk, RSCAN0CFDF0k, and RSCAN0CFDF1k registers and write FF_H to the RSCAN0CFPCTRk register. Thereby the CFMC[7:0] value in the RSCAN0CFSTSk register is decremented by 1 and become 00_H, and the CFEMP flag in the RSCAN0CFSTSk register becomes 1 (the transmit/receive FIFO buffer contains no message (buffer empty)).

(6) When the message matches the receive rule of the corresponding channel and has been successfully received, the routing processing to transfer the message to the specified buffer starts. When the DCE bit in the RSCAN0GCFG register is set to 1 (DLC check is enabled), the DLC filter processing starts at this time.

(7) The message having passed through the DLC filter process is stored in the transmit/receive FIFO buffer which is set to receive mode, when the CFE bit is set to 1 (transmit/receive FIFO buffers are used) and the CFDC[2:0] bits are set to 001_B or more. The CFMC[7:0] value is incremented by 1 to 01H. When the CFIM bit is set to 1 (an interrupt occurs each time a message has been received), the CFRXIF flag is set to 1 (a transmit/receive FIFO reception interrupt request is present). The message is stored in the receive FIFO buffer when the RFE bit in the RSCAN0RFCCx register is set to 1 (receive FIFO buffers are used.) and RFDC[2:0] bits in the RSCAN0RFCCx register are set to 001B or more. The RFMC[7:0] value in theRSCAN0RFSTSx register is incremented by 1 to 01H. When the RFIM bit in the RSCAN0RFCCx register are set to 1 (an interrupt occurs each time a message has been received), the RFIF flag in the RSCAN0RFSTSx register is set to 1 (a receive FIFO interrupt request is present).

16.5.3 Transmission Procedure

16.5.3.1 Procedure for Transmission from Transmit Buffers

Figure 16.26 shows the procedure for transmission from transmit buffers.

Figure 16.27 shows a timing chart where messages are transmitted from two transmit buffers in the same channel and transmission has been successfully completed. **Figure 16.28** shows a timing chart where messages are transmitted from two transmit buffers in the same channel and transmit abort has been completed.

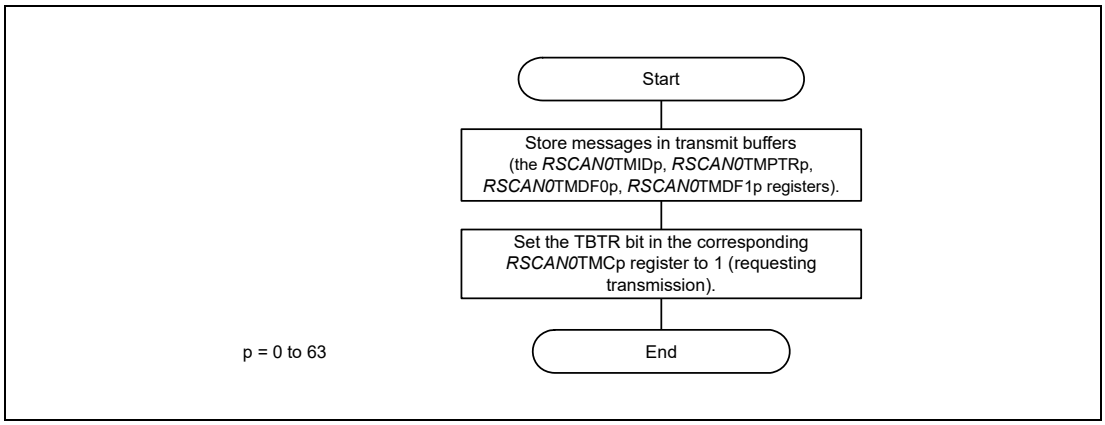


Figure 16.26 Procedure for Transmission from Transmit Buffers

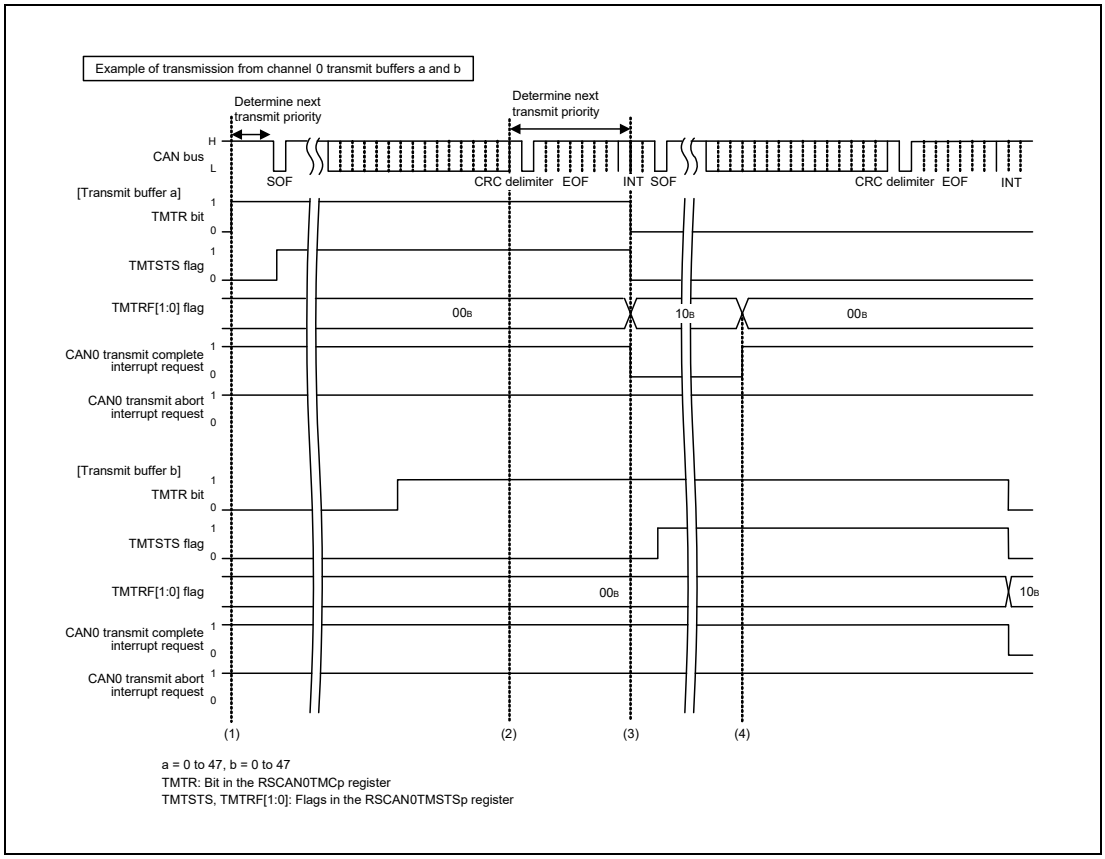


Figure 16.27 Transmit Buffer Transmission Timing Chart (Transmission Completed Successfully)

(1) When the TMTR bit in the RSCAN0TMCa register is set to 1 while the CAN bus is idle, the transmit priority determination processing starts to determine the highest-priority transmit buffer. If transmit buffer a is determined to be the highest-priority transmit buffer, the TMTSTS flag in the corresponding RSCAN0TMSTSa register is set to 1 (transmission is in progress) and the CAN channel starts transmitting data.

(2) When a transmit request from a buffer is present, the priority determination starts with the CRC delimiter for the next transmission. Some delay may occur in the determination time if any other channels are performing the transmit priority determination processing. However, it does not affect the transmission interval because the determination will be completed before bit 3 of intermission.

(3) When transmit completes successfully, the TMTRF[1:0] flag in the RSCAN0TMSTSa register is set to 10_B (transmission has been completed (without transmit abort request)) and the TMTSTS flag and the TMTR bit in the RSCAN0TMSTSa register are cleared to 0. When the TMIEa bit in the RSCAN0TMIEC0 register is 1 (transmit buffer interrupt is enabled), a CAN0 transmit complete interrupt request is generated. To clear the interrupt request, set the TMTRF[1:0] flag to 00_B (transmission is in progress or no transmit request is present).

(4) Before starting the next transmission, set the TMTRF[1:0] flag to 00_B. Write the next message to the transmit buffer, and then set the TMTR bit to 1 (transmission is requested). The TMTR bit can be set to 1 only when the TMTRF[1:0] flag value is 00_B.

If an arbitration lost has occurred after transmission started, the TMTSTS flag is cleared to 0. The transmit priority determination is reexecuted at the beginning of the CRC delimiter to search for the highest-priority transmit buffer. If an error has occurred during transmission or after arbitration lost, the priority determination processing is reexecuted during transmission of an error frame.

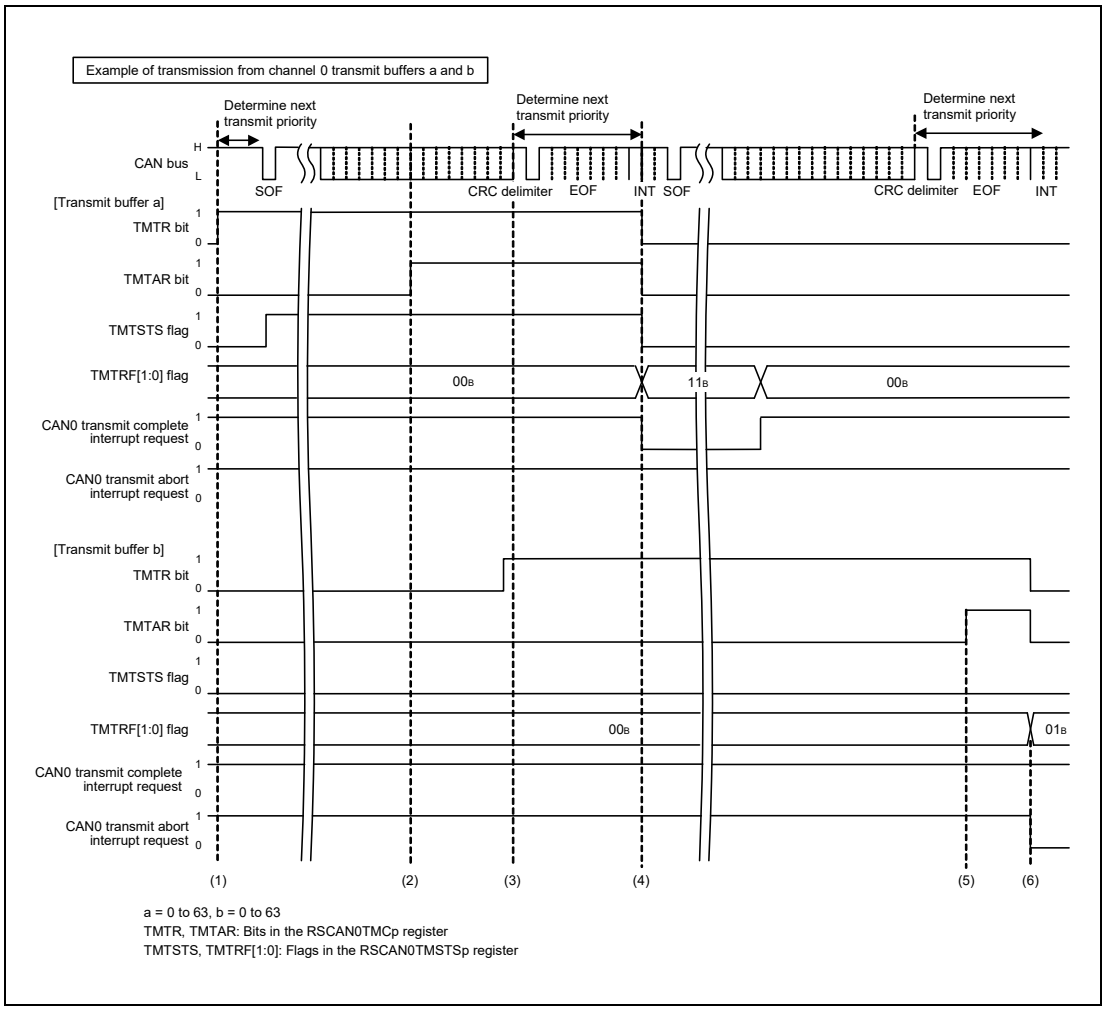


Figure 16.28 Transmit Buffer Transmission Timing Chart (Transmit Abort Completed)

(1) When the TMTR bit in the RSCAN0TMCa register is set to 1 while the CAN bus is idle, the transmit priority determination processing starts to determine the highest-priority transmit buffer. If transmit buffer a is determined to be the highest-priority transmit buffer, the TMTSTS flag in the corresponding RSCAN0TMSTSa register is set to 1 (transmission is in progress) and the CAN channel starts transmitting data.

(2) When the transmit buffer is determined for the next transmission or is in progress of transmission, message transmission is not aborted unless an error or arbitration lost occurs even if the TMTAR bit is set to 1 (transmit abort is requested).

(3) The priority determination starts with the CRC delimiter for the next transmission. In this timing chart, buffer b is not selected as the next transmit buffer. Some delay may occur in the determination time if any other channels are performing the transmit priority determination processing. However, it does not affect the transmission interval because the determination will be completed before bit 3 of intermission.

(4) When transmit completes successfully, the TMTRF[1:0] flag in the RSCAN0TMSTSa register is set to 11_B (transmission has been completed (with transmit abort request)) and the TMTSTS flag and the TMTR bit in the RSCAN0TMCa register are cleared to 0. When the TMIEa bit in the RSCAN0TMIEC0 register is 1 (transmit buffer interrupt is enabled), a CAN0 transmit complete interrupt request is generated. To clear the interrupt request, set the TMTRF[1:0] flag to 00_B (transmission is in progress or no transmit request is present).

(5) When another CAN node is transmitting data on the CAN bus (TMTSTS flag = 0), the TMTR bit cannot be cleared to 0 if the TMTAR bit is set to 1 while the corresponding channel is determining transmit priority.

(6) After the internal processing time has passed, the transmission is terminated and the TMTRF[1:0] flag is set to 01_B. When the transmit buffer is neither transmitting data nor selected as the next transmit buffer and when priority determination is not being made, an abort request is immediately accepted and the TMTRF[1:0] flag is set to 01_B. At this time, the TMTR and TMTAR bits are cleared to 0. When transmit abort is completed with the TAIE bit in the RSCAN0CmCTR register set to 1 (transmit abort interrupt is enabled), an interrupt request is generated. To clear the interrupt request, set the TMTRF[1:0] flag to 00_B.

If an arbitration lost has occurred after the CAN channel started transmission, the TMTSTS bit is cleared to 0. The transmit priority determination is reexecuted at the beginning of the CRC delimiter to search for the highest-priority transmit buffer. If an error has occurred during transmission or after arbitration lost, the priority determination processing is reexecuted during transmission of an error frame.

16.5.3.2 Procedure for Transmission from Transmit/Receive FIFO Buffers

Figure 16.29 shows the procedure for transmission from transmit/receive FIFO buffers.

Figure 16.30 shows a timing chart where messages are transmitted from two transmit/receive FIFO buffers in the same channel and transmission has been successfully completed. **Figure 16.31** shows a timing chart where messages are transmitted from two transmit/receive FIFO buffers in the same channel and transmit abort has been completed.

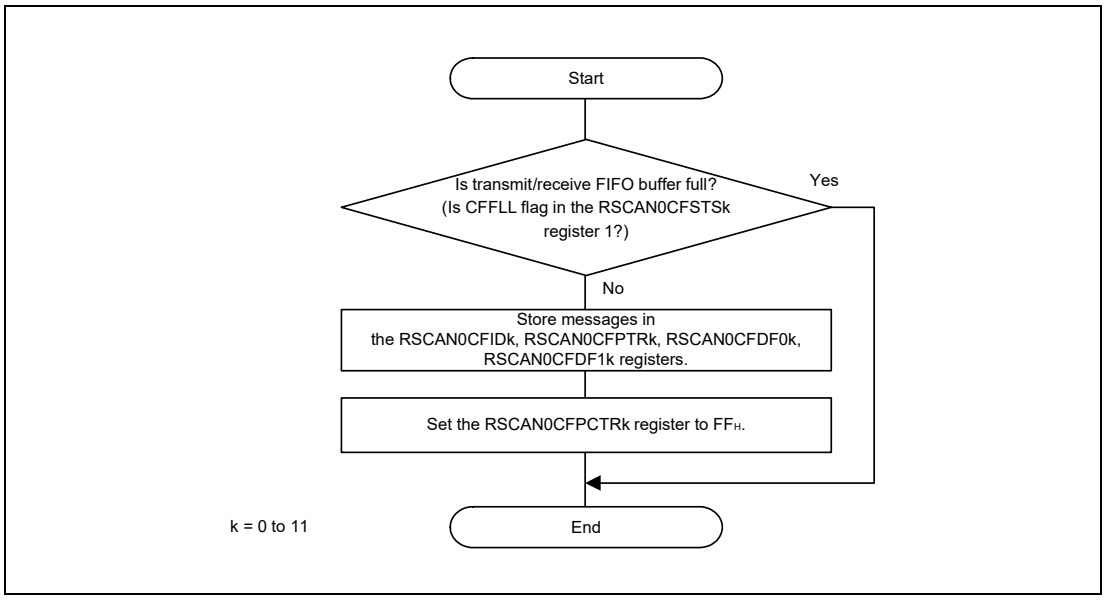


Figure 16.29 Procedure for Transmission from Transmit/Receive FIFO Buffers

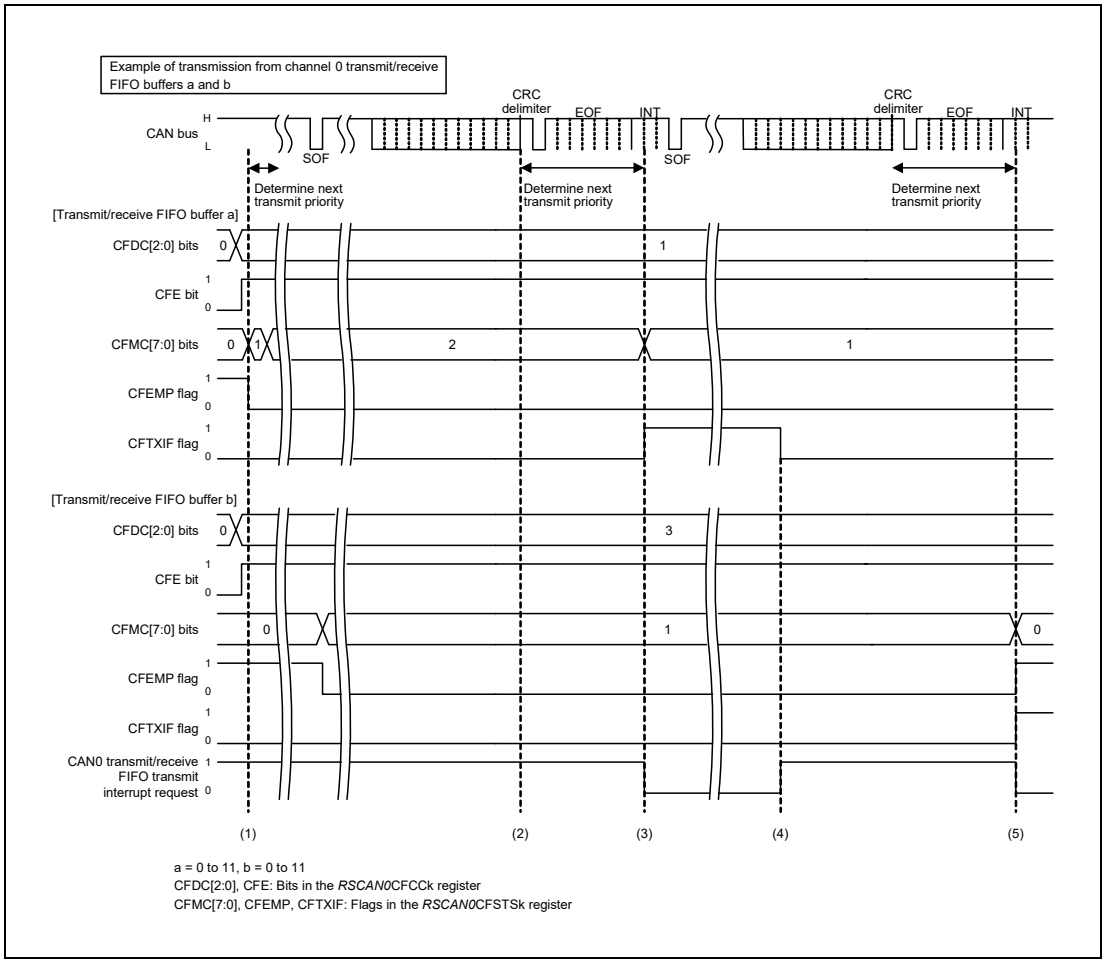


Figure 16.30 Transmit/Receive FIFO Buffer Transmission Timing Chart (Transmission Completed Successfully)

(1) While the CAN bus is idle, the priority determination processing starts to determine the highest-priority transmit message when the CFE bit in the RSCAN0CFCCa register is 1 (transmit/receive FIFO buffers are used), the CFDC[2:0] value in the RSCAN0CFCCa register is 001_B (4 messages) or more, and the CFMC[7:0] value in the RSCAN0CFSTSa register is 01_H or more. When the highest-priority transmit message has been determined, transmission of the message starts. In this figure, the message is transmitted from transmit/receive FIFO buffer a of channel 0.

(2) When a transmit request from a buffer is present, the priority determination starts with the CRC delimiter for the next transmission. Some delay may occur in the determination time if any other channels are performing the transmit priority determination processing. However, it does not affect the transmission interval because the determination will be completed before bit 3 of intermission.

(3) When transmit completes successfully, the CFMC[7:0] value in the RSCAN0CFSTSa register is decremented by 1. Setting the CFIM bit in the RSCAN0CFCCa register to 1 (a FIFO transmission interrupt request is generated each time a message has been transmitted) sets the CFTXIF flag in the RSCAN0CFSTSk register to 1 (a transmit/receive FIFO transmission interrupt request is present).

(4) The CFTXIF flag can be cleared by the program.

(5) Message transmission from transmit/receive FIFO buffer b of channel 0 has been completed and the CFMC[7:0] value in the RSCAN0CFSTSb register is decremented by 1. The CFMC[7:0] bits

are cleared to 00_H and therefore the CFEMP flag in the RSCAN0CFSTSk register is set to 1 (the transmit/receive FIFO buffer contains no message (buffer empty)).

Transmission is continued until the CFEMP flag is set to 1. It is possible to continuously store transmit messages in FIFO buffers until the CFFLL flag in the RSCAN0CFSTSa register and RSCAN0CFSTSb register are set to 1 (the transmit/receive FIFO buffer is full).

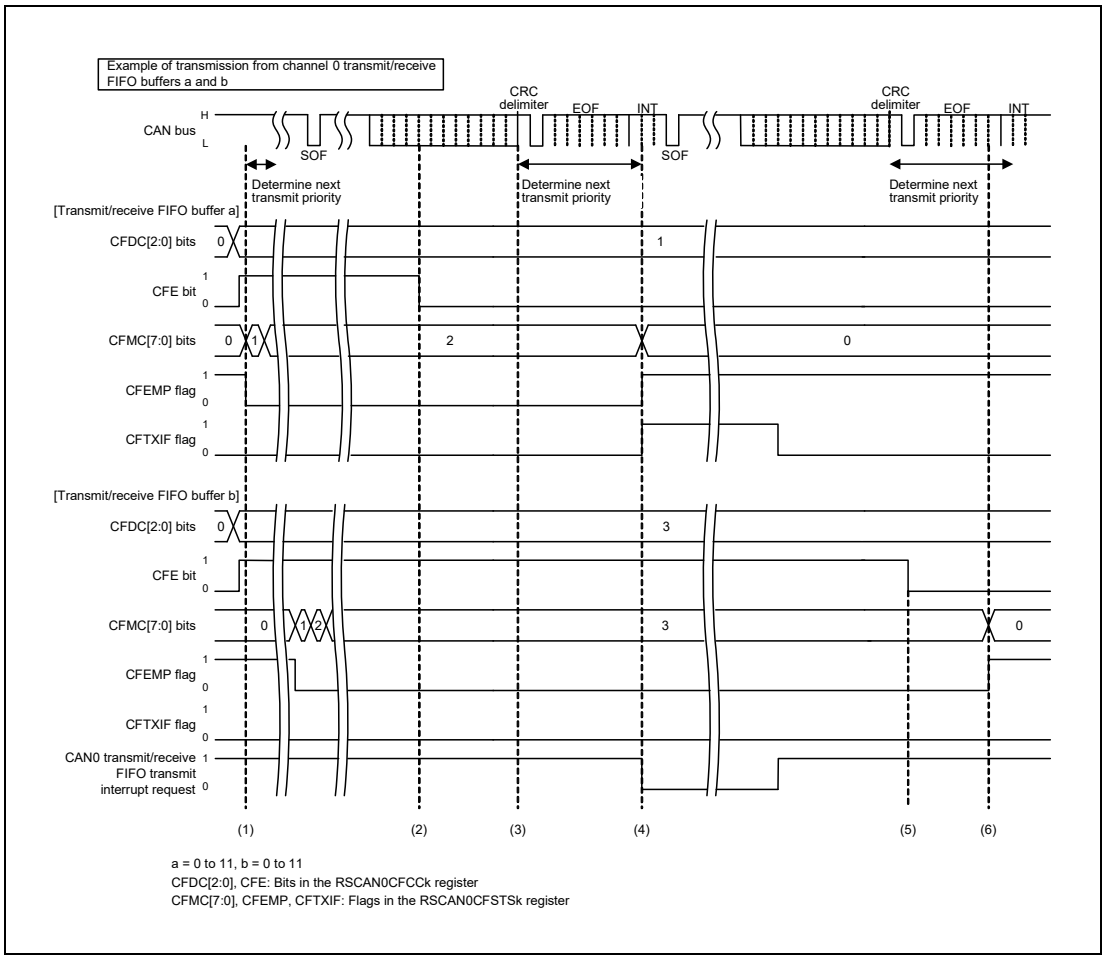


Figure 16.31 Transmit/Receive FIFO Buffer Transmission Timing Chart (Transmit Abort Completed)

(1) While the CAN bus is idle, the priority determination processing starts to determine the highest-priority transmit message when the CFE bit in the RSCAN0CFCCa register (a = 0 to 11) is 1 (transmit/receive FIFO buffers are used), the CFDC[2:0] value in the RSCAN0CFCCa register is 001_B (4 messages) or more, and the CFMC[7:0] value in the RSCAN0CFSTSa register is 01_H or more. When the highest-priority transmit message has been determined, transmission of the message starts. In this figure, the message is transmitted from transmit/receive FIFO buffer a of channel 0.

(2) When transmission is in progress or the transmit/receive FIFO buffer is determined for the next transmission, message transmission is not aborted unless an error or arbitration lost occurs even if the CFE bit is set to 0 (no transmit/receive FIFO buffer is used).

(3) When a transmit request from a buffer is present, the priority determination starts with the CRC delimiter for the next transmission. In this figure, transmit/receive FIFO buffer b is not selected as a buffer for the next transmission. Some delay may occur in the determination time if any other channels are performing the transmit priority determination processing. However, it does not affect the transmission interval because the determination will be completed before bit 3 of intermission.

(4) When transmit completes successfully, the CFMC[7:0] value is cleared to 00_H. Setting the CFIM bit to 1 (a FIFO transmission interrupt request is generated each time a message has been transmitted) sets the CFTXIF flag in the RSCAN0CFSTSa register to 1 (a transmit/receive FIFO transmission interrupt request is present). The CFTXIF flag can be cleared by the program.

(5) If another CAN node on the CAN bus is transmitting data (not from transmit/receive FIFO buffer b), transmit/receive FIFO buffers cannot be disabled immediately even if the CFE bit in the RSCAN0CFCCb register is cleared to 0 (no transmit/receive FIFO buffer is used) during transmit priority determination. (The CFEMP flag in the RSCAN0CFSTSb register is not set to 1 (the transmit/receive FIFO buffer contains no message (buffer empty)) immediately.)

(6) After the internal processing time has passed, transmit/receive FIFO buffers are disabled and the CFMC[7:0] bits in the RSCAN0CFSTSb register are cleared to 00_H and the CFEMP flag is set to 1. When the transmit/receive FIFO buffer is neither transmitting data nor selected as the next transmit buffer and when priority determination is not in progress, the transmit/receive FIFO buffer is immediately disabled. (The CFMC[7:0] bits are cleared to 00_H and the CFEMP flag is set to 1.)

16.5.3.3 Procedure for Transmission from the Transmit Queue

Figure 16.32 shows the procedure for transmission from the transmit queue.

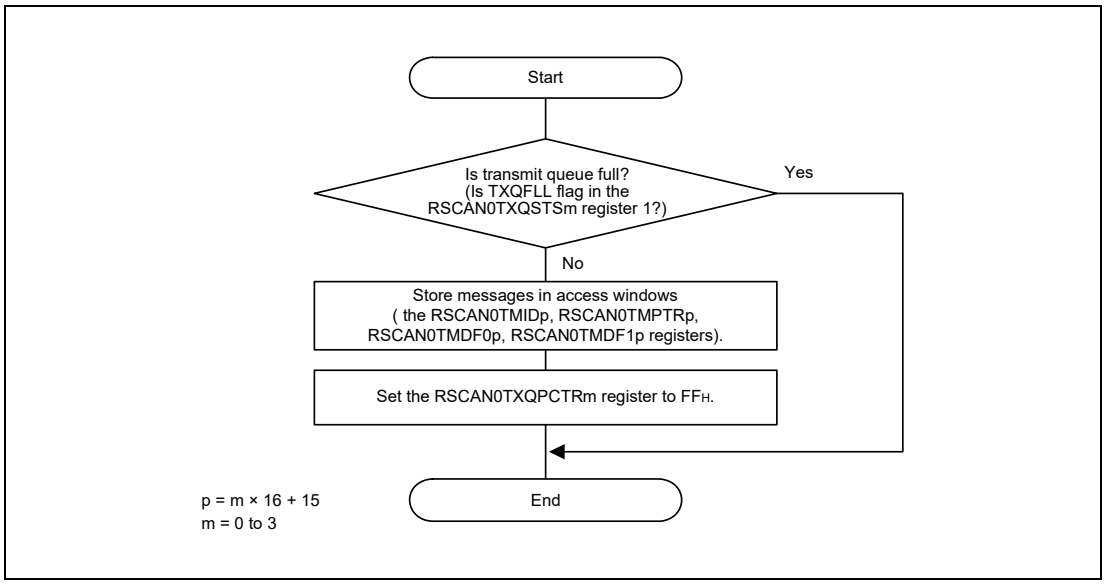


Figure 16.32 Procedure for Transmission from the Transmit Queue

16.5.3.4 Transmit History Buffer Reading Procedure

Transmit history data can be read from the RSCAN0THLACCm register. The next data can be accessed by writing FF_H to the corresponding RSCAN0THLPCTRm register (m = 0 to 3) after reading a set of data. **Figure 16.33** shows the transmit history buffer reading procedure.

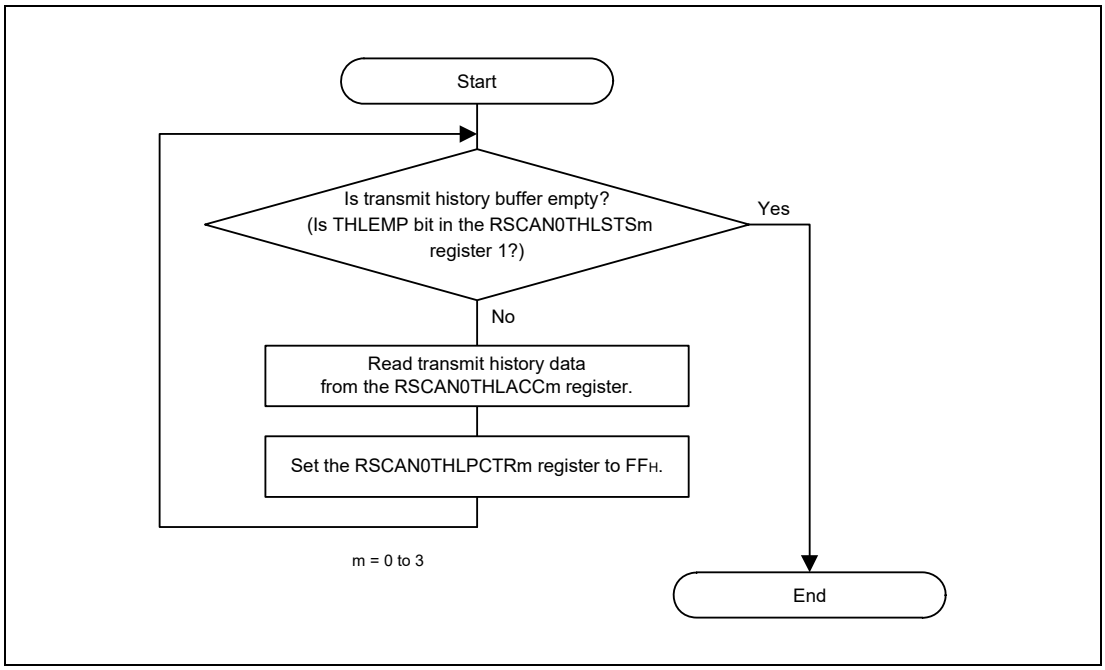


Figure 16.33 Transmit History Buffer Reading Procedure

16.5.4 Test Settings

16.5.4.1 Self-Test Mode Setting Procedure

Self-test mode allows communication test on a channel basis by receiving messages transmitted from the own node.

Figure 16.34 shows the self-test mode setting procedure.

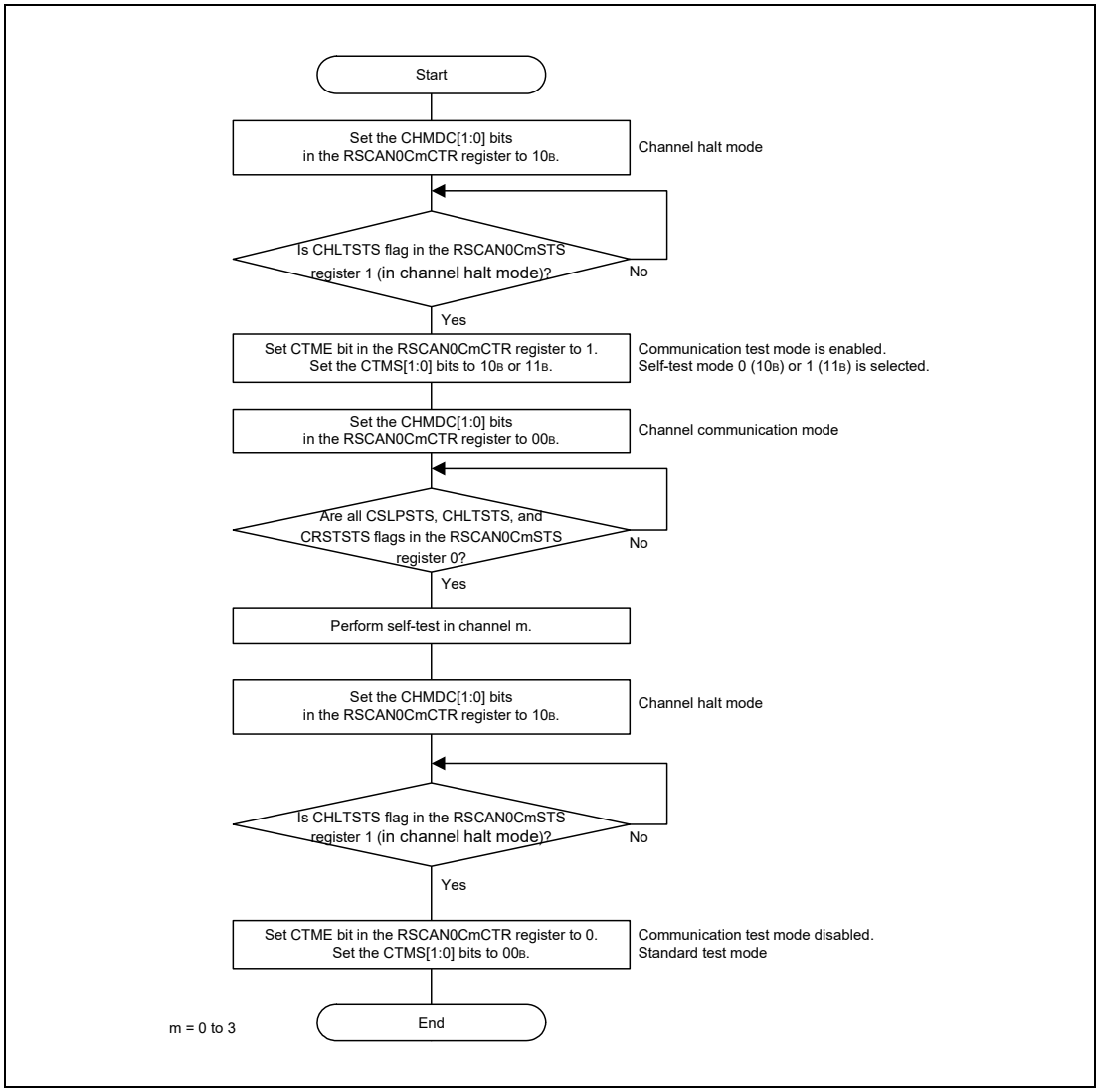


Figure 16.34 Self-Test Mode Setting Procedure

16.5.4.2 Procedure for Releasing the Protection

Since the global test function in **Table 16.94** is protected, write the protection release data 1 and release data 2 in succession to the LOCK[15:0] bits in the RSCAN0GLOCKK register, then set the target test bit to 1.

Table 16.94 Protection Release Data for Test Function

Test Function	Protection Release Data 1	Protection Release Data 2	Target Bit
RAM test	7575_H	$8A8A_H$	RTME bit in the RSCAN0GTSTCTR register

If an incorrect value is written to the LOCK[15:0] bits, restart from writing the protection release data 1. **Figure 16.35** shows the procedure for releasing the protection.

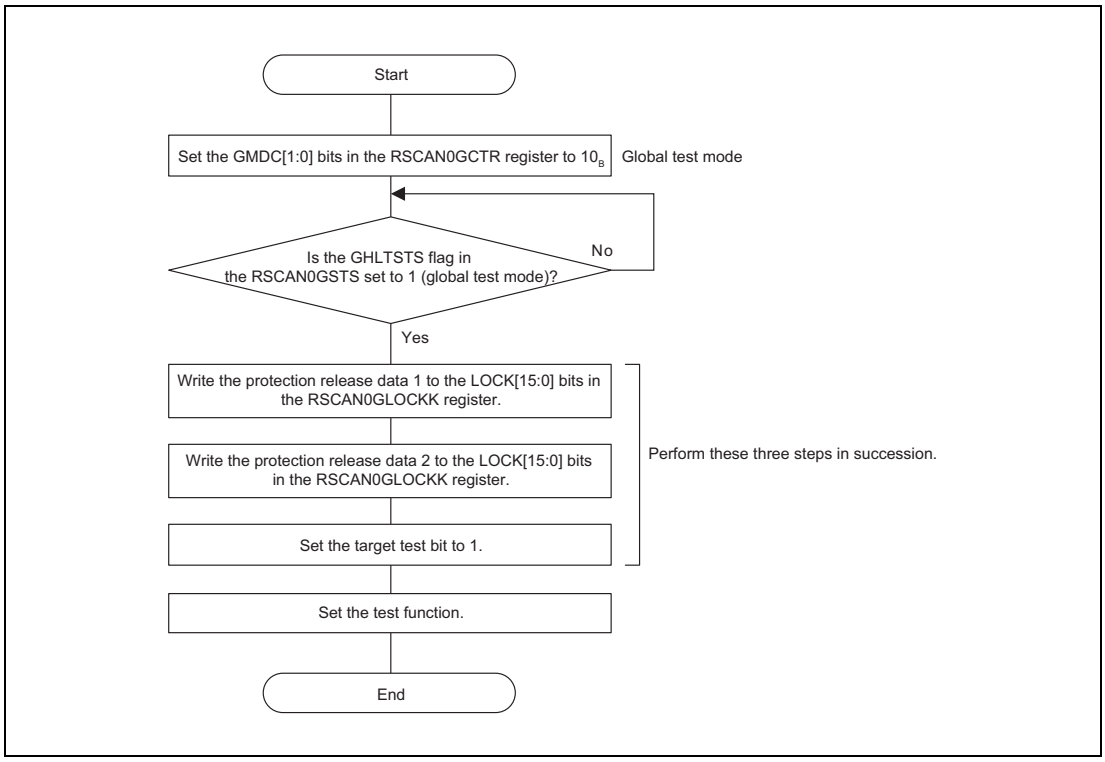


Figure 16.35 Protection Release Procedure

16.5.4.3 RAM Test Setting Procedure

RAM tests include CAN RAM read/write test. The read/write test verifies that data written to the RAM is read correctly. Before closing the RAM test, write 0000 0000_H to all pages of the CAN RAM.

Figure 16.36 shows the RAM test setting procedure.

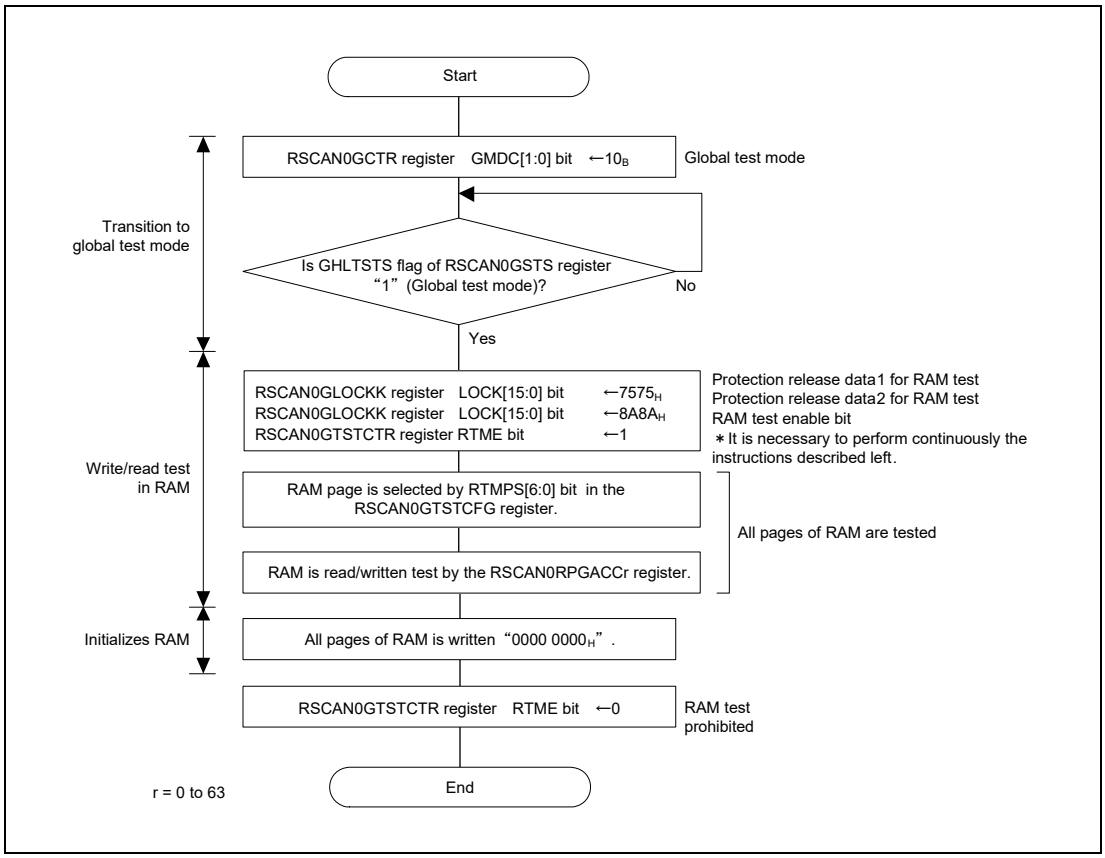


Figure 16.36 RAM Test Setting Procedure

16.5.4.4 Inter-Channel Communication Test Setting Procedure

Communication test can be performed by transmitting and receiving data between different channels.

Figure 16.37 shows the inter-channel communication test setting procedure.

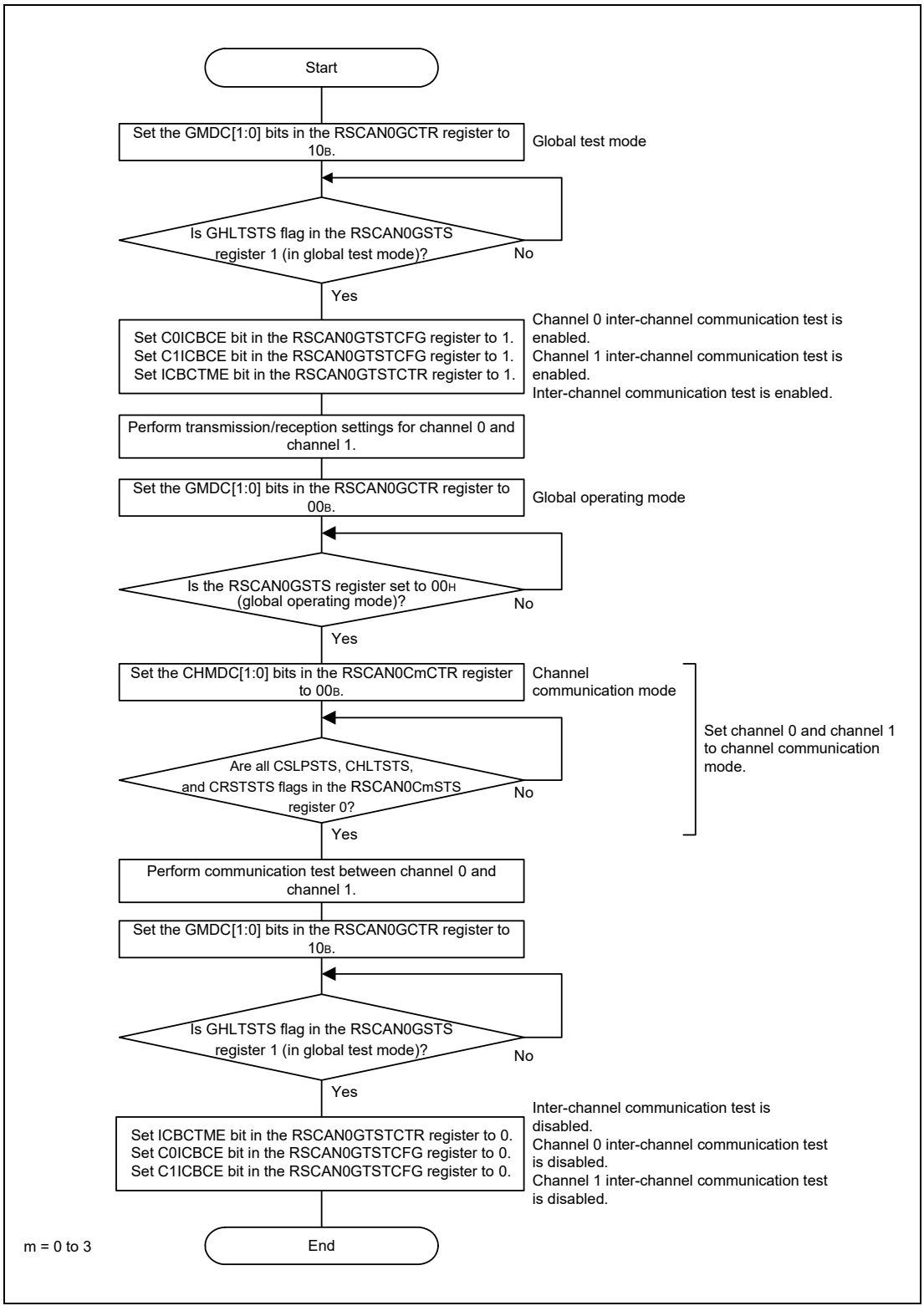


Figure 16.37 Inter-Channel Communication Test Setting Procedure

16.6 Notes on the RS-CAN Module

- When changing a global mode, check the GSLPSTS, GHLTSTS, and GRSTSTS flags in the RSCAN0GSTS register for transitions. When changing a channel mode, check the CSLPSTS, CHLTSTS, and CRSTSTS flags in the RSCAN0CmSTS register (m = 0 to 3) for transitions.
- The acceptance filter processing checks receive rules sequentially in ascending order from the lowest rule number. If the same ID, IDE bit, or RTR bit value is set in multiple receive rules, the lowest-numbered receive rule is used to pass the received message through the acceptance filter processing. If the message does not pass through the subsequent DLC filter processing, the data processing is terminated without returning to the acceptance filter processing and the message is not stored in the buffer.
- When linking transmit buffers to transmit/receive FIFO buffers or allocating transmit buffers to transmit queues, set the control register (RSCAN0TMCp register) of the corresponding transmit buffer to 00_H. The status register (RSCAN0TMSTSp register) of the corresponding transmit buffer should not be used. Flags in other status registers (RSCAN0TMTRSTS0, RSCAN0TMTRSTS1, RSCAN0TMTARSTS0, RSCAN0TMTARSTS1, RSCAN0TMTCSTS0, RSCAN0TMTCSTS1, RSCAN0TMTASTS0, and RSCAN0TMTASTS1 registers), which correspond to transmit buffers linked to transmit/receive FIFO buffers or allocated to transmit queues, remain unchanged. Set the enable bit in the corresponding interrupt enable register (RSCAN0TMIEC0, RSCAN0TMIEC1 register) to 0 (transmit buffer interrupt is disabled).
- Do not allocate transmit buffers that are linked to transmit/receive FIFO buffers to transmit queues.
- Only a single transmit/receive FIFO buffer can be linked to a transmit buffer. Do not link two or more transmit/receive FIFO buffers to the same transmit buffer.
- When the CANm bit time clock is selected as a timestamp counter clock source, the timestamp counter stops when the corresponding channel has transitioned to channel reset mode or channel halt mode.
- If an attempt is made to store a new message when the receive FIFO buffer and the transmit/receive FIFO buffer are full, the new message is discarded. Before attempting to store a new message in the transmit/receive FIFO buffer or the transmit queue, check that the transmit/receive FIFO buffer or the transmit queue is not full.
- The values of unused receive buffers (the RSCAN0RMIDq, RSCAN0RMPTRq, RSCAN0RMDF0q, and RSCAN0RMDF1q registers), receive FIFO buffer access registers (the RSCAN0RFIDx, RSCAN0RFPTRx, RSCAN0RFDF0x, and RSCAN0RFDF1x registers) and transmit/receive FIFO buffer access registers (RSCAN0CFIDk, RSCAN0CFPTRk, RSCAN0CFDF0k, and RSCAN0CFDF1k registers) become undefined once they exit the global reset mode to enter the global operation mode or global test mode.

Section 17 FlexRay

17.1 FlexRay Module

The FlexRay IP-module performs communication according to the FlexRay protocol specification v2.1. With maximum specified sample clock the bitrate is 10 MBit/s. Additional bus driver (BD) hardware is required for connection to the physical layer.

17.1.1 Overview

For communication on a FlexRay network, individual message buffers with up to 254 data bytes are configurable. The message storage consists of a single-ported Message RAM that holds up to 128 message buffers. All functions concerning the handling of messages are implemented in the message handler. Those functions are the acceptance filtering, the transfer of messages between the two FlexRay channel protocol controllers and the message RAM, maintaining the transmission schedule as well as providing message status information.

The register set of the FlexRay IP-module can be accessed directly by an external host via the module's host interface. These registers are used to control/configure/monitor the FlexRay channel protocol controllers, message handler, global time unit, system universal control, frame and symbol processing, network management, interrupt control, to access the message RAM via input/output buffer; and to control the data transfer between the message RAM and the local RAM/global RAM.

The FlexRay IP-module supports the following features:

Item	Specification
Communication	Conformance with FlexRay protocol specification v2.1
Data transfer rate	Up to 10 Mbit/s on each channel
Data link layer clock frequency	80 MHz
Input/output pins per channel	TxD, RxD, ExEN
FlexRay channels	2 (channel A and B)
Message buffer	Up to 128 message buffers configurable
	Configuration of message buffers with different payload lengths possible
	Each message buffer can be configured as receive buffer, as transmit buffer or as part of the receive FIFO
	Filtering for slot counter, cycle counter, and channel
Message RAM	8 Kbytes of message RAM for storage of e.g.
	128 message buffers with max. 48 byte data section
	Up to 30 message buffers with 254 byte data section
FIFO	One configurable receive FIFO
Message buffer access	By host CPU via input and output buffer Input buffer: Holds message to be transferred to the message RAM Output buffer: Holds message read from the message RAM
	By data transfer function Input transfer: Message buffer content is transferred from local RAM/global RAM to message RAM on CPU request Output transfer: Message buffer content is transferred from message RAM to local RAM/global RAM automatically
Network management	Supported
Interrupts	Maskable module interrupts
Timer	Two absolute timers One relative timer One stop watch timer

17.1.2 Terms and Abbreviations

The following terms and abbreviations are used throughout this section.

Abbreviation	Meaning
AP	Action Point
BD	Bus Driver
BSS	Byte Start Sequence
CAS	Collision Avoidance Symbol
CC	Communication Controller
CHI	Controller Host Interface
CIF	Host CPU Interface Block
CRC	Cyclic Redundancy Check
FES	Frame End Sequence
FSS	Frame Start Sequence
FIFO	First In First Out
FSM	Finite State Machine
FSP	Frame and Symbol Processing Block
FTM	Fault Tolerant Midpoint
GIF	Generic Interface Block
GTU	Global Time Unit Block
IBF	Input Buffer
INT	Interrupt Control Block
MBF	Message (Buffer) RAM
MHD	Message Handler Block
MT	Macrotick
MTS	Media Access Test Symbol
NCT	Network Communication Time
NEM	Network Management Block
NIT	Network Idle Time
NM	Network Management
OBF	Output Buffer
PE	Protocol Engine
POC	Protocol Operation Control
PRT	Protocol Controller Block
SDL	Specification and Description Language
SUC	System Universal Control Block
TBF	Transient Buffer
TDMA	Time Division Multiple Access (media access method)
TRH	Transfer Handler
TSS	Transmission Start Sequence
TT-D	Time Triggered Distributed synchronization (protocol mode)
URAM	User RAM (RAM used to store FlexRay messages, message buffers, settings, and states)
µT	Microtick
WUP	Wakeup Pattern
WUS	Wakeup Symbol

17.1.3 Block Diagram

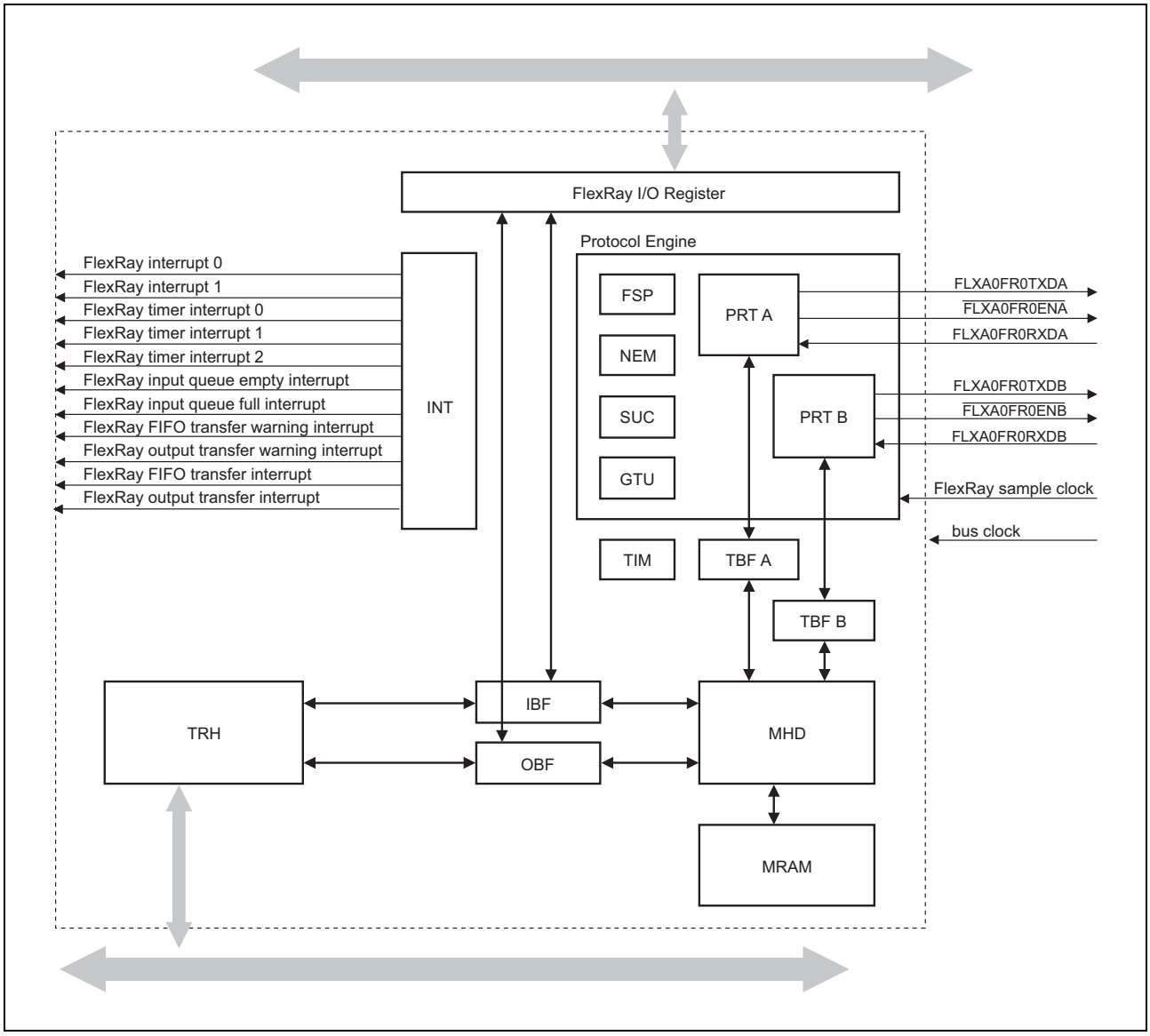


Figure 17.1 FlexRay IP Block Diagram

Input Buffer (IBF)

For write access to the message buffers configured in the message RAM, the host can write the header and data section for a specific message buffer to the input buffer. The message handler then transfers the data from the input buffer to the selected message buffer in the message RAM.

Output Buffer (OBF)

For read access to a message buffer configured in the message RAM the message handler transfers the selected message buffer to the output buffer. After the transfer has completed, the host can read the header and data section of the transferred message buffer from the output buffer.

Message Handler (MHD)

The FlexRay message handler controls data transfers between the following components:

- Input/output buffer and message RAM
- Temporary buffer RAMs of the two FlexRay protocol controllers and message RAM

Message RAM (MRAM)

The message RAM consists of a single-ported RAM that store up to 128 FlexRay message buffers together with the related configuration data (header and data partition).

Temporary Buffer RAM (TBF A/B)

Stores the data section of two complete messages.

FlexRay Channel Protocol Controller (PRT A/B)

The FlexRay channel protocol controllers consist of shift register and FlexRay protocol FSM. They are connected to the temporary buffer RAMs for intermediate message storage and to the physical layer via bus driver BD.

They perform the following functionality:

- Control and check of bit timing
- Reception/transmission of FlexRay frames and symbols
- Check of header CRC
- Generation/checking of frame CRC
- Interfacing to bus driver

Global Time Unit (GTU)

The global time unit performs the following functions:

- Generation of microtick
- Generation of macrotick
- Fault tolerant clock synchronization by FTM algorithm
  - Rate correction
  - Offset correction
- Cycle counter
- Timing control of static segment
- Timing control of dynamic segment (minislotting)
- Support of external clock correction

System Universal Controller(SUC)

The system universal controller controls the following functions:

- Configuration
- Wakeup
- Startup
- Normal Operation
- Passive Operation

Frame and Symbol Processing (FSP)

The frame and symbol processing controls the following functions:

- Checks the correct timing of frames and symbols
- Tests the syntactical and semantical correctness of received frames
- Sets the slot status flags

Network Management (NEM)

Handles the network management vector.

Interrupt Control (INT)

The interrupt controller performs the following functions:

- Provides error and status interrupt flags
- Enable/disable interrupt sources
- Assignment of interrupt sources to one of the two general module interrupt lines
- Enable/disable module interrupt lines

Timer (TIM)

The timer module includes the following macrotick timer:

- Two absolute timers
- One relative timer
- One stop watch timer

Transfer Handler (TRH)

Handles the data transfer between local RAM/global RAM and FlexRay module.

The Transfer Handler supports the following transfer types:

- Transfer of buffer configuration data from the local RAM/global RAM to the message RAM
- Transfer of payload data for transmit buffer from the local RAM/global RAM to the message RAM
- Transfer of buffer configuration data and payload data for transmit buffer from the local RAM/global RAM to the message RAM
- Automatic transfer of payload data from receive buffer to the local RAM/global RAM upon frame reception
- Automatic transfer of payload data, buffer configuration data and message buffer status data from receive buffer to the local RAM/global RAM upon frame reception
- Automatic transfer of buffer configuration data and message buffer status data from the dedicated transmit/receive buffer to the local RAM/global RAM in response to slot status update
- Manual transfer of payload data, buffer configuration data and message buffer status data from the dedicated transmit/receive buffer to the local RAM/global RAM

17.2 Programmer's Model

17.2.1 Register Map

The FlexRay module allocates an address space as shown in **Table 17.2**.

Within this specification the "values after reset" refers to the microcontroller's HW reset. For registers in the address range 0010_H to $0FFF_H$ the "value after reset" is also applicable when the SW reset (using FLXA0FROC.OE) is applied.

The addresses in this specification are listed as offsets from a base address. The base address ERAY must thus be added to the addresses listed in **Table 17.1**. In addition, PE1 accesses the FlexRay registers in a range defined by a different base address (top address in **Table 17.1**) to that for access by the PCU (bottom address in **Table 17.1**). For access from the PCU, "FLXA0" in the register symbol should be replaced by "FLXA0PCU" and the base address becomes that given against "FlexRay (from PCU)" in **Table 17.1**. However, the registers have the same functions.

Table 17.1 Base Address <ERAY>

Base Address	ERAY (User)
FlexRay	$1002\ 0000_H$
FlexRay (from PCU)	$FC02\ 0000_H$

Table 17.2 FlexRay Register Map (1/3)

Register Name	Symbol	Value after Reset	Address	Access Size
FlexRay operation control register	FLXA0FROC	$0000\ 0000_H$	<ERAY> + 0004_H	8, 16, 32
FlexRay operation status register	FLXA0FROS	$0000\ 0000_H$	<ERAY> + $000C_H$	8, 16, 32
FlexRay lock register	FLXA0FRLCK	$0000\ 0000_H$	<ERAY> + $001C_H$	8, 16, 32
FlexRay error interrupt register	FLXA0FREIR	$0000\ 0000_H$	<ERAY> + 0020_H	8, 16, 32
FlexRay status interrupt register	FLXA0FRSIR	$0000\ 0000_H$	<ERAY> + 0024_H	8, 16, 32
FlexRay error interrupt line select	FLXA0FREILS	$0000\ 0000_H$	<ERAY> + 0028_H	8, 16, 32
FlexRay status interrupt line select	FLXA0FRSILS	$0303\ FFFF_H$	<ERAY> + $002C_H$	8, 16, 32
FlexRay error interrupt enable set register	FLXA0FREIES	$0000\ 0000_H$	<ERAY> + 0030_H	8, 16, 32
FlexRay error interrupt enable reset register	FLXA0FREIER	$0000\ 0000_H$	<ERAY> + 0034_H	8, 16, 32
FlexRay status interrupt enable set register	FLXA0FRSIES	$0000\ 0000_H$	<ERAY> + 0038_H	8, 16, 32
FlexRay status interrupt enable reset register	FLXA0FRSIER	$0000\ 0000_H$	<ERAY> + $003C_H$	8, 16, 32
FlexRay interrupt line enable register	FLXA0FRILE	$0000\ 0000_H$	<ERAY> + 0040_H	8, 16, 32
FlexRay timer 0 configuration register	FLXA0FRT0C	$0000\ 0000_H$	<ERAY> + 0044_H	8, 16, 32
FlexRay timer 1 configuration register	FLXA0FRT1C	$0002\ 0000_H$	<ERAY> + 0048_H	8, 16, 32
FlexRay stop watch register 1	FLXA0FRSTPW1	$0000\ 0000_H$	<ERAY> + $004C_H$	8, 16, 32
FlexRay stop watch register 2	FLXA0FRSTPW2	$0000\ 0000_H$	<ERAY> + 0050_H	8, 16, 32
FlexRay SUC configuration register 1	FLXA0FRSUCC1	$0C40\ 1080_H$	<ERAY> + 0080_H	8, 16, 32
FlexRay SUC configuration register 2	FLXA0FRSUCC2	$0100\ 0504_H$	<ERAY> + 0084_H	8, 16, 32
FlexRay SUC configuration register 3	FLXA0FRSUCC3	$0000\ 0011_H$	<ERAY> + 0088_H	8, 16, 32
FlexRay NEM configuration register	FLXA0FRNEMC	$0000\ 0000_H$	<ERAY> + $008C_H$	8, 16, 32
FlexRay PRT configuration register 1	FLXA0FRPRTC1	$084C\ 0633_H$	<ERAY> + 0090_H	8, 16, 32
FlexRay PRT configuration register 2	FLXA0FRPRTC2	$0F2D\ 0A0E_H$	<ERAY> + 0094_H	8, 16, 32
FlexRay MHD configuration register	FLXA0FRMHDC	$0000\ 0000_H$	<ERAY> + 0098_H	8, 16, 32
FlexRay GTU configuration register 1	FLXA0FRGTUC1	$0000\ 0280_H$	<ERAY> + $00A0_H$	8, 16, 32

Table 17.2 FlexRay Register Map (2/3)

Register Name	Symbol	Value after Reset	Address	Access Size
FlexRay GTU configuration register 2	FLXA0FRGTUC2	0002 000A$_H$	<ERAY> + 00A4$_H$	8, 16, 32
FlexRay GTU configuration register 3	FLXA0FRGTUC3	0202 0000$_H$	<ERAY> + 00A8$_H$	8, 16, 32
FlexRay GTU configuration register 4	FLXA0FRGTUC4	0008 0007$_H$	<ERAY> + 00AC$_H$	8, 16, 32
FlexRay GTU configuration register 5	FLXA0FRGTUC5	0E00 0000$_H$	<ERAY> + 00B0$_H$	8, 16, 32
FlexRay GTU configuration register 6	FLXA0FRGTUC6	0002 0000$_H$	<ERAY> + 00B4$_H$	8, 16, 32
FlexRay GTU configuration register 7	FLXA0FRGTUC7	0002 0004$_H$	<ERAY> + 00B8$_H$	8, 16, 32
FlexRay GTU configuration register 8	FLXA0FRGTUC8	0000 0002$_H$	<ERAY> + 00BC$_H$	8, 16, 32
FlexRay GTU configuration register 9	FLXA0FRGTUC9	0000 0101$_H$	<ERAY> + 00C0$_H$	8, 16, 32
FlexRay GTU configuration register 10	FLXA0FRGTUC10	0002 0005$_H$	<ERAY> + 00C4$_H$	8, 16, 32
FlexRay GTU configuration register 11	FLXA0FRGTUC11	0000 0000$_H$	<ERAY> + 00C8$_H$	8, 16, 32
FlexRay CC status vector register	FLXA0FRCCSV	0010 4000$_H$	<ERAY> + 0100$_H$	8, 16, 32
FlexRay CC error vector register	FLXA0FRCCEV	0000 0000$_H$	<ERAY> + 0104$_H$	8, 16, 32
FlexRay slot counter value register	FLXA0FRSCV	0000 0000$_H$	<ERAY> + 0110$_H$	8, 16, 32
FlexRay macrotick and cycle counter value register	FLXA0FRMTCCV	0000 0000$_H$	<ERAY> + 0114$_H$	8, 16, 32
FlexRay rate correction value register	FLXA0FRRCV	0000 0000$_H$	<ERAY> + 0118$_H$	8, 16, 32
FlexRay offset correction value register	FLXA0FROCV	0000 0000$_H$	<ERAY> + 011C$_H$	8, 16, 32
FlexRay sync frame status register	FLXA0FRSFS	0000 0000$_H$	<ERAY> + 0120$_H$	8, 16, 32
FlexRay symbol window and NIT status register	FLXA0FRSWNIT	0000 0000$_H$	<ERAY> + 0124$_H$	8, 16, 32
FlexRay aggregated channel status register	FLXA0FRACS	0000 0000$_H$	<ERAY> + 0128$_H$	8, 16, 32
FlexRay even sync ID register n (n=1 to 15)	FLXA0FRESIDn (n = 1 to 15)	0000 0000$_H$	<ERAY> + 0130$_H$ to <ERAY> + 0168$_H$ (<ERAY> + 0130$_H$ + (n-1)*4)	8, 16, 32
FlexRay odd sync ID register n (n=1 to 15)	FLXA0FROSIDn (n = 1 to 15)	0000 0000$_H$	<ERAY> + 0170$_H$ to <ERAY> + 01A8$_H$ (<ERAY> + 0170$_H$ + (n-1)*4)	8, 16, 32
FlexRay network management vector register n (n = 1 to 3)	FLXA0FRNMVn (n = 1 to 3)	0000 0000$_H$	<ERAY> + 01B0$_H$ to <ERAY> + 01B8$_H$ (<ERAY> + 01B0$_H$ + (n-1)*4)	8, 16, 32
FlexRay message RAM configuration register	FLXA0FRMRC	0180 0000$_H$	<ERAY> + 0300$_H$	8, 16, 32
FlexRay FIFO rejection filter register	FLXA0FRFRF	0180 0000$_H$	<ERAY> + 0304$_H$	8, 16, 32
FlexRay FIFO rejection filter mask register	FLXA0FRFRFM	0000 0000$_H$	<ERAY> + 0308$_H$	8, 16, 32
FlexRay FIFO critical level register	FLXA0FRFCL	0000 0080$_H$	<ERAY> + 030C$_H$	8, 16, 32
FlexRay message handler status register	FLXA0FRMHDS	0000 0080$_H$	<ERAY> + 0310$_H$	8, 16, 32
FlexRay last dynamic transmit slot register	FLXA0FRLDTS	0000 0000$_H$	<ERAY> + 0314$_H$	8, 16, 32
FlexRay FIFO status register	FLXA0FRFSR	0000 0000$_H$	<ERAY> + 0318$_H$	8, 16, 32
FlexRay message handler constraints flags register	FLXA0FRMHDF	0000 0000$_H$	<ERAY> + 031C$_H$	8, 16, 32
FlexRay transmission register i (i = 1 to 4)	FLXA0FRTXRQi (i = 1 to 4)	0000 0000$_H$	<ERAY> + 0320$_H$ to <ERAY> + 032C$_H$ (<ERAY> + 0320 $_H$ + (i-1)*4)	8, 16, 32
FlexRay new data register i (i = 1 to 4)	FLXA0FRNDATi (i = 1 to 4)	0000 0000$_H$	<ERAY> + 0330$_H$ to <ERAY> + 033C$_H$ (<ERAY> + 0330$_H$ + (i-1)*4)	8, 16, 32
FlexRay message buffer status changed register i (i = 1 to 4)	FLXA0FRMBSCi (i = 1 to 4)	0000 0000$_H$	<ERAY> + 0340$_H$ to <ERAY> + 034C$_H$ (<ERAY> + 0340$_H$ + (i-1)*4)	8, 16, 32

Table 17.2 FlexRay Register Map (3/3)

Register Name	Symbol	Value after Reset	Address	Access Size
FlexRay write data section register n (n = 1 to 64)	FLXA0FRWRDSn (n = 1 to 64)	0000 0000_H	<ERAY> + 0400_H to <ERAY> + $04FC_H$ (<ERAY> + 0400_H + (n-1)*4)	8, 16, 32
FlexRay write header section register 1	FLXA0FRWRHS1	0000 0000_H	<ERAY> + 0500_H	8, 16, 32
FlexRay write header section register 2	FLXA0FRWRHS2	0000 0000_H	<ERAY> + 0504_H	8, 16, 32
FlexRay write header section register 3	FLXA0FRWRHS3	0000 0000_H	<ERAY> + 0508_H	8, 16, 32
FlexRay input buffer command mask register	FLXA0FRIBCM	0000 0000_H	<ERAY> + 0510_H	8, 16, 32
FlexRay input buffer command request register	FLXA0FRIBCR	0000 0000_H	<ERAY> + 0514_H	8, 16, 32
FlexRay read data section register n (n = 1 to 64)	FLXA0FRRDDSn (n = 1 to 64)	0000 0000_H	<ERAY> + 0600_H to <ERAY> + $06FC_H$ (<ERAY> + 0600_H + (n-1)*4)	8, 16, 32
FlexRay read header section register 1	FLXA0FRRDHS1	0000 0000_H	<ERAY> + 0700_H	8, 16, 32
FlexRay read header section register 2	FLXA0FRRDHS2	0000 0000_H	<ERAY> + 0704_H	8, 16, 32
FlexRay read header section register 3	FLXA0FRRDHS3	0000 0000_H	<ERAY> + 0708_H	8, 16, 32
FlexRay message buffer status register	FLXA0FRMBS	0000 0000_H	<ERAY> + $070C_H$	8, 16, 32
FlexRay output buffer command mask register	FLXA0FROBCM	0000 0000_H	<ERAY> + 0710_H	8, 16, 32
FlexRay output buffer command request register	FLXA0FROBCR	0000 0000_H	<ERAY> + 0714_H	8, 16, 32
FlexRay input transfer configuration register	FLXA0FRITC	0000 0000_H	<ERAY> + 0800_H	8, 16, 32
FlexRay output transfer configuration register	FLXA0FROTC	0000 0000_H	<ERAY> + 0804_H	8, 16, 32
FlexRay input pointer table base address register	FLXA0FRIBA	0000 0000_H	<ERAY> + 0808_H	8, 16, 32
FlexRay FIFO pointer table base address register	FLXA0FRFBA	0000 0000_H	<ERAY> + $080C_H$	8, 16, 32
FlexRay Output pointer table base address register	FLXA0FROBA	0000 0000_H	<ERAY> + 0810_H	8, 16, 32
FlexRay input queue control register	FLXA0FRIQC	0000 0000_H	<ERAY> + 0814_H	8, 16, 32
FlexRay user input transfer request register	FLXA0FRUIR	0000 0000_H	<ERAY> + 0818_H	8, 16, 32
FlexRay user output transfer request register	FLXA0FRUOR	0000 0000_H	<ERAY> + $081C_H$	8, 16, 32
FlexRay input transfer status register	FLXA0FRITS	0000 0000_H	<ERAY> + 0820_H	8, 16, 32
FlexRay output transfer status register	FLXA0FROTS	0000 0000_H	<ERAY> + 0824_H	8, 16, 32
FlexRay access error status register	FLXA0FRAES	0000 0000_H	<ERAY> + 0828_H	8, 16, 32
FlexRay access error address register	FLXA0FRAEA	0000 0000_H	<ERAY>+ $082C_H$	8, 16, 32
FlexRay message data available register i (i = 0 to 3)	FLXA0FRDAi (i = 0 to 3)	0000 0000_H	<ERAY> + 0830_H to <ERAY> + $083C_H$ (<ERAY> + 0830_H + i*4)	8, 16, 32
FlexRay H-Bus configuration register	FLXA0FRAHBC	0000 0000_H	<ERAY> + 0840_H	8, 16, 32
FlexRay timer 2 configuration register	FLXA0FRT2C	0000 0000_H	<ERAY> + 0844_H	8, 16, 32

17.2.2 FlexRay Operation Register

17.2.2.1 FLXA0FROC — FlexRay Operation Control Register

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: ERAY + 0004_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	T2IE	T1IE	T0IE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	OEP	—	—	—	—	—	BEC	OE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R/W	R/W

Table 17.3 FLXA0FROC Register Contents

Bit Position	Bit Name	Function
31 to 19	—	Reserved These bits are always read as 0. When writing, always write 0.
18	T2IE	Timer 2 Interrupt Enable 0: Disabled 1: Enabled
17	T1IE	Timer 1 Interrupt Enable 0: Disabled 1: Enabled
16	T0IE	Timer 0 Interrupt Enable 0: Disabled 1: Enabled
15 to 8	—	Reserved These bits are always read as 0. When writing, always write 0.
7	OEP	Operation Enable Bit Protection 0: OE bit is unprotected 1: OE bit is protected
6 to 2	—	Reserved These bits are always read as 0. When writing, always write 0.
1	BEC	Byte Endian Control 0: Little endian 1: Big endian
0	OE	Operation Enable 0: Operation stopped SW reset 1: Operation Enabled

(1) FLXA0FROC.T2IE

Timer 2 Interrupt enable

This bit controls the timer 2 interrupt.

0: Disabled

No interrupt will be requested and the timer 2 interrupt line will be released if pending.

1: Enabled

Timer 2 interrupt will be asserted when FLXA0FROTS.T2IS is ‘1’.

(2) FLXA0FROC.T1IE

Timer 1 Interrupt enable

The user should only set this bit to ‘1’ when timer 1 interrupt is not enabled in the FlexRay Status interrupt enable register (FLXA0FRSIES.TI1E should be ‘0’).

This bit controls the timer 1 interrupt.

0: Disabled

No interrupt will be requested and the timer 1 interrupt line will be released if pending.

1: Enabled

Timer 1 interrupt will be asserted when FLXA0FROTS.T1IS is ‘1’.

(3) FLXA0FROC.T0IE

Timer 0 Interrupt enable

The user should only set this bit to ‘1’ when timer 0 interrupt is not enabled in the FlexRay Status interrupt enable register (FLXA0FRSIES.TI0E should be ‘0’).

This bit controls the timer 0 interrupt.

0: Disabled

No interrupt will be requested and the timer 0 interrupt line will be released if pending.

1: Enabled

Timer 0 interrupt will be asserted when FLXA0FROTS.T0IS is ‘1’.

(4) FLXA0FROC.OEP

Operation Enable Bit Protection

This bit protects against unintended write access to the OE bit.

0: OE bit is unprotected

Write access to the OE bit is enabled

1: OE bit is protected

Write access to the OE bit is disabled

(5) FLXA0FROC.BEC

Byte Endian Control

The user should only change this bit when FLXA0FROS.OS is ‘1’.

This bit controls the byte order on reading and writing the FlexRay network management vector register (FLXA0FRNMVn), FlexRay write data section (FLXA0FRWRDSn) and FlexRay read data section (FLXA0FRRDDSn). This bit also controls the byte order when reading or writing FlexRay payload data using the data transfer function.

For details about the byte alignment please refer to **Section 17.3.17, Byte Alignment**.

0: Little endian

Byte alignment in FLXA0FRNMVn, FLXA0FRWRDSn and FLXA0FRRDDSn is in little endian style.

1: Big endian

Byte alignment in FLXA0FRNMVn, FLXA0FRWRDSn and FLXA0FRRDDSn is in big endian style.

(6) FLXA0FROC.OE

Operation Enable

The user can only write to this bit when FLXA0FROC.OEP is ‘0’.

The user should only write this bit with ‘0’ when FLXA0FROS.OS is ‘1’.

The user should only write this bit with ‘1’ when FLXA0FROS.OS is ‘0’ and the FlexRay sample clock is enabled.

This bit controls the operation state and serves the software reset of the FlexRay module. The operation status bit (FLXA0FROS.OS) indicates whether the FlexRay module is in reset state or not.

0: Operation stopped SW reset

Forcibly moves the FlexRay module to its reset state, whatever the state of the FlexRay module is.

1: Operation Enabled

Reset state of the FlexRay module is released.

17.2.2.2 FLXA0FROS — FlexRay Operation Status Register

Do not rewrite this register using bit-manipulation instruction.

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: ERAY + $000C_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	T2IS	T1IS	T0IS
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	OS
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 17.4 FLXA0FROS Register Contents

Bit Position	Bit Name	Function
31 to 19	—	Reserved These bits are always read as 0. When writing, always write 0.
18	T2IS	Timer 2 Interrupt Status 0: Timer 2 has not matched the conditions configured in the FLXA0FRT2C register 1: Timer 2 matched the conditions configured in the FLXA0FRT2C register
17	T1IS	Timer 1 Interrupt Status 0: Timer 1 has not matched the conditions configured in the FLXA0FRT1C register 1: Timer 1 matched the conditions configured in the FLXA0FRT1C register
16	T0IS	Timer 0 Interrupt Status 0: Timer 0 has not matched the conditions configured in the FLXA0FRTOC register 1: Timer 0 matched the conditions configured in the FLXA0FRT0C register
15 to 1	—	Reserved These bits are always read as 0. When writing, always write 0.
0	OS	Operation Status 0: Operation disabled, reset state 1: Operation enabled

(1) FLXA0FROS.T2IS

Timer 2 Interrupt Status

Writing ‘0’ has no effect on the bit value.

This bit represents that Timer 2 has matched the state configured in FLXA0FRT2C register.

If enabled in FLXA0FROC.T2IE the timer 2 interrupt is generated when FLXA0FROS.T2IS is ‘1’.

[Clearing condition]

This bit is cleared by writing ‘1’ to FLXA0FROS.T2IS.

This bit is cleared when FLXA0FROS.OS changes from ‘1’ to ‘0’.

[Setting condition]

This bit is set when Timer 2 matches the state configured in FLXA0FRT2C register.

(2) FLXA0FROS.T1IS

Timer 1 Interrupt Status

Writing ‘0’ has no effect on the bit value.

This bit represents that Timer 1 has matched the state configured in FLXA0FRT1C register.

If enabled in FLXA0FROC.T1IE the timer 1 interrupt is generated when FLXA0FROS.T1IS is ‘1’.

[Clearing condition]

This bit is cleared by writing ‘1’ to FLXA0FROS.T1IS.

This bit is cleared when FLXA0FROS.OS changes from ‘1’ to ‘0’.

[Setting condition]

This bit is set when Timer 1 matches the state configured in FLXA0FRT1C register.

(3) FLXA0FROS.T0IS

Timer 0 Interrupt Status

Writing ‘0’ has no effect on the bit value.

This bit represents that the Timer 0 has matched the state configured in FLXA0FRT0C register.

If enabled in FLXA0FROC.T0IE the timer 1 interrupt is generated when FLXA0FROS.T0IS is ‘1’.

[Clearing condition]

This bit is cleared by writing ‘1’ to FLXA0FROS.T0IS.

This bit is cleared when FLXA0FROS.OS changes from ‘1’ to ‘0’.

[Setting condition]

This bit is set when Timer 0 matches the state configured in FLXA0FRT0C register.

(4) FLXA0FROS.OS

Operation Status

This bit represents if the FlexRay module is in the reset or the operation state.

When FLXA0FROS.OS is ‘0’ the FlexRay module gets initialized and registers mapped to the address area ERAY + 0010_H to ERAY + $0FFF_H$ cannot be accessed; read access from these registers will return undefined data.

When FLXA0FROS.OS is ‘1’ it is possible to access to the address area ERAY + 0010_H to ERAY + $0FFF_H$ and to perform FlexRay communication.

When FLXA0FROS.OS changes from ‘0’ to ‘1’ all registers in the address range ERAY + 0010_H to ERAY + $0FFF_H$ are set to the “Values after reset”.

[Clearing condition]

When FLXA0FROC.OE is set to ‘0’, It takes up to two peripheral bus clock cycles until FLXA0FROS.OS is set to ‘0’.

[Setting condition]

When FLXA0FROC.OE is set to ‘1’ it takes up to four peripheral clock cycles of the clock with the lower frequency out of the FlexRay sample clock and peripheral bus clock until FLXA0FROS.OS is set to ‘1’.

17.2.3 Special Registers

17.2.3.1 FLXA0FRLCK — FlexRay Lock Register

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: ERAY + $001C_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	CLK[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 17.5 FLXA0FRLCK Register Contents

Bit	Symbol	Function
31 to 8	—	Reserved These bits are always read as 0. When writing, always write 0.
7 to 0	CLK[7:0]	Configuration Lock Key

(1) FLXA0FRLCK.CLK

Configuration Lock Key

The lock register is write-only. Reading the register will return $0000\ 0000_H$.

To leave CONFIG state by writing FLXA0FRSUCC1.CMD (command READY), the write operation has to be directly preceded by two write accesses to the Configuration Lock Key (unlock sequence). If the write sequence below is interrupted by other write accesses between the second write to the configuration lock key and the write access to the FLXA0FRSUCC1 register, the CC remains in CONFIG state and the sequence has to be repeated.

First write:FLXA0FRLCK.CLK = “1100 1110” (CE_H)

Second write:FLXA0FRLCK.CLK = “0011 0001” (31_H)

Third write:FLXA0FRSUCC1.CMD

NOTE

In case that the host uses 8/16-bit accesses to write the listed bit fields, the programmer has to ensure that no “dummy accesses” e.g. to the remaining register bytes/words are inserted by the compiler.

17.2.4 Interrupt Registers

17.2.4.1 FLXA0FREIR — FlexRay Error Interrupt Register

Do not rewrite this register using bit-manipulation instruction.

The flags are set when the CC detects one of the listed error conditions. The flags remain set until the host clears them.

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: ERAY + 0020_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	TABB	LTVB	EDB	—	—	—	—	—	TABA	LTVA	EDA
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	MHF	IOBA	IIBA	EFA	RFO	AERR	CCL	CCF	SFO	SFBM	CNA	PEMC
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 17.6 FLXA0FREIR Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 27	—	Reserved These bits are always read as 0. When writing, always write 0.
26	TABB	Transmission Across Boundary Channel B Flag 0: No transmission across slot boundary detected on channel B 1: Transmission across slot boundary detected on channel B
25	LTVB	Latest Transmit Violation Channel B Flag 0: No latest transmit violation detected on channel B 1: Latest transmit violation detected on channel B
24	EDB	Error Detected on Channel B Flag 0: No error detected on channel B 1: Error detected on channel B
23 to 19	—	Reserved These bits are always read as 0. When writing, always write 0.
18	TABA	Transmission Across Boundary Channel A Flag 0: No transmission across slot boundary detected on channel A 1: Transmission across slot boundary detected on channel A
17	LTVA	Latest Transmit Violation Channel A Flag 0: No latest transmit violation detected on channel A 1: Latest transmit violation detected on channel A
16	EDA	Error Detected on Channel A Flag 0: No error detected on channel A 1: Error detected on channel A
15 to 12	—	Reserved These bits are always read as 0. When writing, always write 0.
11	MHF	Message Handler Constraints Flag 0: No message handler failure detected 1: Message handler failure detected

Table 17.6 FLXA0FREIR Register Contents (2/2)

Bit Position	Bit Name	Function
10	IOBA	Illegal Output Buffer Access Flag 0: No illegal host access to output buffer occurred 1: Illegal host access to output buffer occurred
9	IIBA	Illegal Input Buffer Access Flag 0: No illegal host access to input buffer occurred 1: Illegal host access to input buffer occurred
8	EFA	Empty FIFO Access Flag 0: No host access to empty FIFO occurred 1: Host access to empty FIFO occurred
7	RFO	Receive FIFO Overrun Flag 0: No receive FIFO overrun detected 1: A receive FIFO overrun has been detected
6	AERR	Access Error Flag 0: Access error is not detected. 1: Access error is detected.
5	CCL	CHI Command Locked Flag 0: CHI command accepted 1: CHI command not accepted
4	CCF	Clock Correction Failure Flag 0: No clock correction error 1: Clock correction failed
3	SFO	Sync Frame Overflow Flag 0: Number of received sync frames <= FLXA0FRGTUC2.SNM 1: More sync frames received than configured by FLXA0FRGTUC2.SNM
2	SFBM	Sync Frames Below Minimum Flag 0: Sync node: 1 or more sync frames received, non-sync node: 2 or more sync frames received 1: Less than the required minimum of sync frames received
1	CNA	Command Not Accepted Flag 0: CHI command accepted 1: CHI command not accepted
0	PEMC	POC Error Mode Changed Flag 0: Error mode has not changed 1: Error mode has changed

(1) FLXA0FREIR.TABB

Transmission Across Boundary Channel B Flag

Writing ‘0’ has no effect on the bit value.

This bit is cleared when writing ‘1’ to it.

The flag signals to the host that a transmission across a slot boundary occurred for channel B.

(2) FLXA0FREIR.LTVB

Latest Transmit Violation Channel B Flag

Writing ‘0’ has no effect on the bit value.

This bit is cleared when writing ‘1’ to it.

The flag signals a latest transmit violation on channel B to the host.

(3) FLXA0FREIR.EDB

Error Detected on Channel B Flag

Writing ‘0’ has no effect on the bit value.

This bit is cleared when writing ‘1’ to it.

This bit is set whenever one of the flags FLXA0FRACS.SEDB, FLXA0FRACS.CEDB, FLXA0FRACS.CIB, FLXA0FRACS.SBVB changes from ‘0’ to ‘1’.

(4) FLXA0FREIR.TABA

Transmission Across Boundary Channel A Flag

Writing ‘0’ has no effect on the bit value.

This bit is cleared when writing a ‘1’ to it.

The flag signals to the host that a transmission across a slot boundary occurred for channel A.

(5) FLXA0FREIR.LTVA

Latest Transmit Violation Channel A Flag

Writing ‘0’ has no effect on the bit value.

This bit is cleared when writing a ‘1’ to it.

The flag signals a latest transmit violation on channel A to the host.

(6) FLXA0FREIR.EDA

Error Detected on Channel A Flag

Writing ‘0’ has no effect on the bit value.

This bit is cleared when writing a ‘1’ to it.

This bit is set whenever one of the flags FLXA0FRACS.SEDA, FLXA0FRACS.CEDA, FLXA0FRACS.CIA, FLXA0FRACS.SBVA changes from ‘0’ to ‘1’.

(7) FLXA0FREIR.MHF

Message Handler Constraints Flag

Writing ‘0’ has no effect on the bit value.

This bit is cleared when writing a ‘1’ to it.

The flag signals a message handler constraints violation condition. It is set whenever one of the flags FLXA0FRMHDF.SNUA, FLXA0FRMHDF.SNUB, FLXA0FRMHDF.FNFA, FLXA0FRMHDF.FNFB, FLXA0FRMHDF.TBFA, FLXA0FRMHDF.TBFB, FLXA0FRMHDF.WAHP changes from ‘0’ to ‘1’.

(8) FLXA0FREIR.IOBA

Illegal Output Buffer Access Flag

Writing ‘0’ has no effect on the bit value.

This bit is cleared when writing a ‘1’ to it.

This flag is set by the CC when the host requests the transfer of a message buffer from the message RAM to the output buffer while FLXA0FROBCR.OBSYS is set to ‘1’.

(9) FLXA0FREIR.IIBA

Illegal Input Buffer Access Flag

Writing ‘0’ has no effect on the bit value.

This bit is cleared when writing a ‘1’ to it.

This flag is set by the CC when the host wants to modify a message buffer via input buffer and one of the following conditions applies:

1. The CC is not in CONFIG or DEFAULT_CONFIG state and the host writes to the input buffer command request register to modify the
   - Header section of message buffer 0, 1 if configured for transmission in key slot
   - Header section of static message buffers with buffer number < FLXA0FRMRC.FDB while FLXA0FRMRC.SEC = “01”
   - Header section of any static or dynamic message buffer while FLXA0FRMRC.SEC = “1x”
   - Header and/or data section of any message buffer belonging to the receive FIFO
2. The host writes to any register of the input buffer while FLXA0FRIBCR.IBSYH is set to ‘1’.

(10) FLXA0FREIR.EFA

Empty FIFO Access Flag

Writing ‘0’ has no effect on the bit value.

This bit is cleared when writing a ‘1’ to it.

This flag is set by the CC when the host requests the transfer of a message from the receive FIFO via output buffer while the receive FIFO is empty.

(11) FLXA0FREIR.RFO

Receive FIFO Overrun Flag

Writing ‘0’ has no effect on the bit value.

This bit is cleared when writing a ‘1’ to it.

The flag is set by the CC when a receive FIFO overrun is detected. When a receive FIFO overrun occurs, the oldest message is overwritten with the actual received message. The actual state of the FIFO is monitored in register FLXA0FRFSR.

(12) FLXA0FREIR.AERR

Access Error flag

Writing 0 in this bit has no effect.

This bit is cleared when writing ‘1’ to it.

Notifies of an access error.

When the AMR, ATBF1, or ATBF2 bit in the FLXA0FRMHDS register changes from 0 to 1, this bit is set to 1.

(13) FLXA0FREIR.CCL

CHI Command Locked Flag

Writing ‘0’ has no effect on the bit value.

This bit is cleared when writing a ‘1’ to it.

The flag signals that the write access to the CHI command vector FLXA0FRSUCC1.CMD was not successful because the execution of the previous CHI command has not yet completed. In this case bit FLXA0FREIR.CNA is also set to ‘1’.

(14) FLXA0FREIR.CCF

Clock Correction Failure Flag

Writing ‘0’ has no effect on the bit value.

This bit is cleared when writing a ‘1’ to it.

This flag is set at the end of the communication cycle whenever one of the following errors occurred:

- Missing offset and/or rate correction
- Clock correction limit reached

The clock correction status is monitored in registers FLXA0FRCCEV and FLXA0FRSFS. A failure may occur during startup, therefore bit FLXA0FREIR.CCF should be set to “0” after the CC entered NORMAL_ACTIVE state.

(15) FLXA0FREIR.SFO

Sync Frame Overflow Flag

Writing ‘0’ has no effect on the bit value.

This bit is cleared when writing a ‘1’ to it.

It is set to “1” when either the number of sync frames received during the last communication cycle or the total number of different sync frame IDs received during the last double cycle exceeds the maximum number of sync frames as defined by FLXA0FRGTUC2.SNM.

(16) FLXA0FREIR.SFBM

Sync Frames Below Minimum Flag

Writing ‘0’ has no effect on the bit value.

This bit is cleared when writing a ‘1’ to it.

This flag is set to “1” at the end of a cycle if the number of sync frames received during the last communication cycle was below the limit required for rate or offset correction term calculation (i.e. missing offset and/or missing rate correction). The clock correction status is monitored in FLXA0FRCCEV and FLXA0FRSFS.

This flag may be set to “1” during startup. Therefore this flag should be set to “0” by the host after the CC entered NORMAL_ACTIVE state.

(17) FLXA0FREIR.CNA

Command Not Accepted Flag

Writing ‘0’ has no effect on the bit value.

This bit is cleared when writing a ‘1’ to it.

The flag signals that the write access to the CHI command vector FLXA0FRSUCC1.CMD was not successful because the requested command was not valid in the actual POC state, or because the CHI command was locked (CCL = ‘1’).

(18) FLXA0FREIR.PEMC

POC Error Mode Changed Flag

Writing ‘0’ has no effect on the bit value.

This bit is cleared when writing a ‘1’ to it.

This flag is set to “1” whenever the error mode signaled by FLXA0FRCCEV.ERRM has changed.

17.2.4.2 FLXA0FRSIR — FlexRay Status Interrupt Register

Do not rewrite this register using bit-manipulation instruction.

The flags are set when the CC detects one of the listed events. The flags remain set until the host clears them.

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: ERAY + 0024_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	MTSB	WUPB	—	—	—	—	—	—	MTSA	WUPA
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SDS	MBSI	SUCS	SWE	TOBC	TIBC	TI1	TI0	NMVC	RFCL	RFNE	RXI	TXI	CYCS	CAS	WST
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 17.7 FLXA0FRSIR Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 26	—	Reserved These bits are always read as 0. When writing, always write 0.
25	MTSB	MTS Received on Channel B Flag (vSS!ValidMTSB) 0: No MTS symbol received on channel B 1: MTS symbol received on channel B
24	WUPB	Wakeup Pattern Channel B Flag 0: No wakeup pattern received on channel B 1: Wakeup pattern received on channel B
23 to 18	—	Reserved These bits are always read as 0. When writing, always write 0.
17	MTSA	MTS Received on Channel A Flag (vSS!ValidMTSA) 0: No MTS symbol received on channel A 1: MTS symbol received on channel A
16	WUPA	Wakeup Pattern Channel A Flag 0: No wakeup pattern received on channel A 1: Wakeup pattern received on channel A
15	SDS	Start of Dynamic Segment Flag 0: Dynamic segment not yet started 1: Dynamic segment started
14	MBSI	Message Buffer Status Interrupt Flag 0: No message buffer status change of message buffer with MBI = ‘1’ 1: Message buffer status of at least one message buffer with MBI = ‘1’ has changed
13	SUCS	Startup Normal End Flag 0: No startup completed successfully 1: Startup completed successfully
12	SWE	Stop Watch Event Flag 0: No Stop Watch Event 1: Stop Watch Event occurred
11	TOBC	Transfer Output Buffer Completed Flag 0: No transfer completed 1: Transfer between message RAM and output buffer completed

Table 17.7 FLXA0FRSIR Register Contents (2/2)

Bit Position	Bit Name	Function
10	TIBC	Transfer Input Buffer Completed Flag 0: No transfer completed 1: Transfer between input buffer and message RAM completed
9	TI1	Timer Interrupt 1 Flag 0: No timer interrupt 1 1: Timer interrupt 1 occurred
8	TI0	Timer Interrupt 0 Flag 0: No timer interrupt 0 1: Timer interrupt 0 occurred
7	NMVC	Network Management Vector Changed Flag 0: No change in the network management vector 1: Network management vector changed
6	RFCL	Receive FIFO Critical Level Flag 0: Receive FIFO below critical level 1: Receive FIFO critical level reached
5	RFNE	Receive FIFO Not Empty Flag 0: Receive FIFO is empty 1: Receive FIFO is not empty
4	RXI	Receive Interrupt Flag 0: No ND flag of a receive buffer with MBI = ‘1’ has been set to ‘1’ 1: At least one ND flag of a receive buffer with MBI = 1 has been set to 1
3	TXI	Transmit Interrupt Flag 0: No frame transmitted from a transmit buffer with MBI = ‘1” 1: At least one frame was transmitted from a transmit buffer with MBI = ‘1’
2	CYCS	Cycle Start Interrupt Flag 0: No communication cycle started 1: Communication cycle started
1	CAS	Collision Avoidance Symbol Flag 0: No bit pattern matching the CAS symbol received 1: Bit pattern matching the CAS symbol received
0	WST	Wakeup Status Flag 0: Wakeup status unchanged 1: Wakeup status changed

(1) FLXA0FRSIR.MTSB

MTS Received on Channel B Flag (vSS!ValidMTSB)

Writing ‘0’ has no effect on the bit value.

This bit is cleared when writing a ‘1’ to it.

Media Access Test Symbol (MTS) received on channel B during the preceding symbol window.

Updated by the CC for each channel at the end of the symbol window.

(2) FLXA0FRSIR.WUPB

Wakeup Pattern Channel B Flag

Writing ‘0’ has no effect on the bit value.

This bit is cleared when writing a ‘1’ to it.

This flag is set to “1” when a wakeup pattern was received on channel B in either of the following states:

- WAKEUP
- READY
- STARTUP

(3) FLXA0FRSIR.MTSA

MTS Received on Channel A Flag (vSS!ValidMTSA)

Writing ‘0’ has no effect on the bit value.

This bit is cleared when writing a ‘1’ to it.

Media Access Test Symbol (MTS) received on channel A during the preceding symbol window.

Updated by the CC for each channel at the end of the symbol window.

(4) FLXA0FRSIR.WUPA

Wakeup Pattern Channel A Flag

Writing ‘0’ has no effect on the bit value.

This bit is cleared when writing a ‘1’ to it.

This flag is set to “1” when a wakeup pattern was received on channel A in either of the following states:

- WAKEUP
- READY
- STARTUP

(5) FLXA0FRSIR.SDS

Start of Dynamic Segment Flag

Writing ‘0’ has no effect on the bit value.

This bit is cleared when writing a ‘1’ to it.

This flag is set by the CC when the dynamic segment starts.

(6) FLXA0FRSIR.MBSI

Message Buffer Status Interrupt Flag

Writing ‘0’ has no effect on the bit value.

This bit is cleared when writing a ‘1’ to it.

This flag is set by the CC when the message buffer status FLXA0FRMBS has changed and if bit MBI of that message buffer is “1” (see **Table 17.100**).

(7) FLXA0FRSIR.SUCS

Startup Normal End Flag

Writing ‘0’ has no effect on the bit value.

This bit is cleared when writing a ‘1’ to it.

This flag is set whenever a startup completed successfully and the CC entered NORMAL_ACTIVE state.

(8) FLXA0FRSIR.SWE

Stop Watch Event Flag

Writing ‘0’ has no effect on the bit value.

This bit is cleared when writing a ‘1’ to it.

This flag is set after a stop watch activation when the actual cycle counter and macrotick value are stored in the Stop Watch register (see **Section 17.2.5.4, FLXA0FRSTPW1 — FlexRay Stop Watch Register 1**).

(9) FLXA0FRSIR.TOBC

Transfer Output Buffer Completed Flag

Writing ‘0’ has no effect on the bit value.

This bit is cleared when writing a ‘1’ to it.

This flag is set whenever a transfer from the message RAM to the output buffer has completed and FLXA0FROBCR.OBSYS has been reset by the message handler.

(10) FLXA0FRSIR.TIBC

Transfer Input Buffer Completed Flag

Writing ‘0’ has no effect on the bit value.

This bit is cleared when writing a ‘1’ to it.

This flag is set whenever a transfer from input buffer to the message RAM has completed and FLXA0FRIBCR.IBSYS has been reset by the message handler.

(11) FLXA0FRSIR.TI1

Timer Interrupt 1 Flag

Writing ‘0’ has no effect on the bit value.

This bit is cleared when writing a ‘1’ to it.

This flag is set whenever timer 1 matches the conditions configured in register FLXA0FRT1C.

FlexRay timer interrupt 1 is generated when the T1IE bit in the FLXA0FROC register is effective.

(12) FLXA0FRSIR.TI0

Timer Interrupt 0 Flag

Writing ‘0’ has no effect on the bit value.

This bit is cleared when writing a ‘1’ to it.

This flag is set whenever timer 0 matches the conditions configured in register FLXA0FRT0C.

FlexRay timer interrupt 0 is generated when the T0IE bit in the FLXA0FROC register is effective.

(13) FLXA0FRSIR.NMVC

Network Management Vector Changed Flag

Writing ‘0’ has no effect on the bit value.

This bit is cleared when writing a ‘1’ to it.

This is set when a change in the Network Management Vector occurs.

(14) FLXA0FRSIR.RFCL

Receive FIFO Critical Level Flag

Writing ‘0’ has no effect on the bit value.

This bit is cleared when writing a ‘1’ to it.

This flag is set when the receive FIFO fill level FLXA0FRFSR.RFFL is equal or greater than the critical level as configured by FLXA0FRFCL.CL.

(15) FLXA0FRSIR.RFNE

Receive FIFO Not Empty Flag

Writing ‘0’ has no effect on the bit value.

This bit is cleared when writing a ‘1’ to it.

This flag is set when a received valid frame was stored into the empty receive FIFO. The current state of the receive FIFO can be read in register FLXA0FRFSR.

(16) FLXA0FRSIR.RXI

Receive Interrupt Flag

Writing ‘0’ has no effect on the bit value.

This bit is cleared when writing a ‘1’ to it.

This flag is set whenever the set condition of a message buffers ND flag is fulfilled (see **Section 17.2.9.6, FLXA0FRNDATi — FlexRay New Data Register i (i = 1 to 4)**), and if bit MBI of that message buffer is set to 1 (see **Table 17.100**).

(17) FLXA0FRSIR.TXI

Transmit Interrupt Flag

Writing ‘0’ has no effect on the bit value.

This bit is cleared when writing a ‘1’ to it.

This flag is set at the end of frame transmission if bit MBI in the respective message buffer is set to 1 (see **Table 17.100**).

(18) FLXA0FRSIR.CYCS

Cycle Start Interrupt Flag

Writing ‘0’ has no effect on the bit value.

This bit is cleared when writing a ‘1’ to it.

This flag is set to “1” every time a communication cycle starts.

(19) FLXA0FRSIR.CAS

Collision Avoidance Symbol Flag

Writing ‘0’ has no effect on the bit value.

This bit is cleared when writing a ‘1’ to it.

This flag is set during STARTUP state when a CAS or an expected CAS was received.

(20) FLXA0FRSIR.WST

Wakeup Status Flag

Writing ‘0’ has no effect on the bit value.

This bit is cleared when writing a ‘1’ to it.

This flag is set when FLXA0FRCCSV.WSV changes to a value other than UNDEFINED.

17.2.4.3 FLXA0FREILS — FlexRay Error Interrupt Line Select Register

The FlexRay error interrupt line select register assigns an interrupt generated by a specific error interrupt flag from register FLXA0FREIR to one of the two modules interrupt lines (FlexRay interrupt 0 or FlexRay interrupt 1).

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: ERAY + 0028_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	TABBL	LTVBL	EDBL	—	—	—	—	—	TABAL	LTVAL	EDAL
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	MHFL	IOBAL	IBAL	EFAL	RFOL	AERRL	CCLL	CCFL	SFOL	SFBML	CNAL	PEMCL
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 17.8 FLXA0FREILS Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 27	—	Reserved These bits are always read as 0. When writing, always write 0.
26	TABBL	Transmission Across Boundary Channel B Interrupt Line 0: Interrupt assigned to FlexRay interrupt 0 1: Interrupt assigned to FlexRay interrupt 1
25	LTVBL	Latest Transmit Violation Channel B Interrupt Line 0: Interrupt assigned to FlexRay interrupt 0 1: Interrupt assigned to FlexRay interrupt 1
24	EDBL	Error Detected on Channel B Interrupt Line 0: Interrupt assigned to FlexRay interrupt 0 1: Interrupt assigned to FlexRay interrupt 1
23 to 19	—	Reserved These bits are always read as 0. When writing, always write 0.
18	TABAL	Transmission Across Boundary Channel A Interrupt Line 0: Interrupt assigned to FlexRay interrupt 0 1: Interrupt assigned to FlexRay interrupt 1
17	LTVAL	Latest Transmit Violation Channel A Interrupt Line 0: Interrupt assigned to FlexRay interrupt 0 1: Interrupt assigned to FlexRay interrupt 1
16	EDAL	Error Detected on Channel A Interrupt Line 0: Interrupt assigned to FlexRay interrupt 0 1: Interrupt assigned to FlexRay interrupt 1
15 to 12	—	Reserved These bits are always read as 0. When writing, always write 0.
11	MHFL	Message Handler Constraints Flag Interrupt Line 0: Interrupt assigned to FlexRay interrupt 0 1: Interrupt assigned to FlexRay interrupt 1
10	IOBAL	Illegal Output Buffer Access Interrupt Line 0: Interrupt assigned to FlexRay interrupt 0 1: Interrupt assigned to FlexRay interrupt 1

Table 17.8 FLXA0FREILS Register Contents (2/2)

Bit Position	Bit Name	Function
9	IIBAL	Illegal Input Buffer Access Interrupt Line 0: Interrupt assigned to FlexRay interrupt 0 1: Interrupt assigned to FlexRay interrupt 1
8	EFAL	Empty FIFO Access Interrupt Line 0: Interrupt assigned to FlexRay interrupt 0 1: Interrupt assigned to FlexRay interrupt 1
7	RFOL	Receive FIFO Overrun Interrupt Line 0: Interrupt assigned to FlexRay interrupt 0 1: Interrupt assigned to FlexRay interrupt 1
6	AERRL	Access Error Interrupt Output Select 0: Interrupt assigned to FlexRay interrupt request 0 1: Interrupt assigned to FlexRay interrupt request 1
5	CCLL	CHI Command Locked Interrupt Line 0: Interrupt assigned to FlexRay interrupt 0 1: Interrupt assigned to FlexRay interrupt 1
4	CCFL	Clock Correction Failure Interrupt Line 0: Interrupt assigned to FlexRay interrupt 0 1: Interrupt assigned to FlexRay interrupt 1
3	SFOL	Sync Frame Overflow Interrupt Line 0: Interrupt assigned to FlexRay interrupt 0 1: Interrupt assigned to FlexRay interrupt 1
2	SFBML	Sync Frames Sync Frame Shortfall Interrupt Output Selecting 0: Interrupt assigned to FlexRay interrupt 0 1: Interrupt assigned to FlexRay interrupt 1
1	CNAL	Command Not Accepted Interrupt Line 0: Interrupt assigned to FlexRay interrupt 0 1: Interrupt assigned to FlexRay interrupt 1
0	PEMCL	POC Error Mode Changed Interrupt Line 0: Interrupt assigned to FlexRay interrupt 0 1: Interrupt assigned to FlexRay interrupt 1

17.2.4.4 FLXA0FRSILS — FlexRay Status Interrupt Line Select Register

The FlexRay status interrupt line select register assign an interrupt generated by a specific status interrupt flag from register FLXA0FRSIR to one of the two module interrupt lines (FlexRay interrupt 0, FlexRay interrupt 1).

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: ERAY + $002C_H$

Value after reset: $0303\ FFFF_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	MTSBL	WUPBL	—	—	—	—	—	—	MTSAL	WUPAL
Value after reset	0	0	0	0	0	0	1	1	0	0	0	0	0	0	1	1
R/W	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SDSL	MBSIL	SUCSL	SWEL	TOBCL	TIBCL	TI1L	TI0L	NMVCL	RFCLL	RFNEL	RXIL	TXIL	CYCSL	CASL	WSTL
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 17.9 FLXA0FRSILS Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 26	—	Reserved These bits are always read as 0. When writing, always write 0.
25	MTSBL	Media Access Test Symbol Channel B Interrupt Line 0: Interrupt assigned to FlexRay interrupt 0 1: Interrupt assigned to FlexRay interrupt 1
24	WUPBL	Wakeup Pattern Channel B Interrupt Line 0: Interrupt assigned to FlexRay interrupt 0 1: Interrupt assigned to FlexRay interrupt 1
23 to 18	—	Reserved These bits are always read as 0. When writing, always write 0.
17	MTSAL	Media Access Test Symbol Channel A Interrupt Line 0: Interrupt assigned to FlexRay interrupt 0 1: Interrupt assigned to FlexRay interrupt 1
16	WUPAL	Wakeup Pattern Channel A Interrupt Line 0: Interrupt assigned to FlexRay interrupt 0 1: Interrupt assigned to FlexRay interrupt 1
15	SDSL	Start of Dynamic Segment Interrupt Line 0: Interrupt assigned to FlexRay interrupt 0 1: Interrupt assigned to FlexRay interrupt 1
14	MBSIL	Message Buffer Status Interrupt Line 0: Interrupt assigned to FlexRay interrupt 0 1: Interrupt assigned to FlexRay interrupt 1
13	SUCSL	Startup Completed Successfully Interrupt Line 0: Interrupt assigned to FlexRay interrupt 0 1: Interrupt assigned to FlexRay interrupt 1
12	SWEL	Stop Watch Event Interrupt Line 0: Interrupt assigned to FlexRay interrupt 0 1: Interrupt assigned to FlexRay interrupt 1
11	TOBCL	Transfer Output Buffer Completed Interrupt Line 0: Interrupt assigned to FlexRay interrupt 0 1: Interrupt assigned to FlexRay interrupt 1

Table 17.9 FLXA0FRSILS Register Contents (2/2)

Bit Position	Bit Name	Function
10	TIBCL	Transfer Input Buffer Completed Interrupt Line 0: Interrupt assigned to FlexRay interrupt 0 1: Interrupt assigned to FlexRay interrupt 1
9	TI1L	Timer Interrupt 1 Line 0: Interrupt assigned to FlexRay interrupt 0 1: Interrupt assigned to FlexRay interrupt 1
8	TI0L	Timer Interrupt 0 Line 0: Interrupt assigned to FlexRay interrupt 0 1: Interrupt assigned to FlexRay interrupt 1
7	NMVCL	Network Management Vector Changed Interrupt Line 0: Interrupt assigned to FlexRay interrupt 0 1: Interrupt assigned to FlexRay interrupt 1
6	RFCLL	Receive FIFO Critical Level Interrupt Line 0: Interrupt assigned to FlexRay interrupt 0 1: Interrupt assigned to FlexRay interrupt 1
5	RFNEL	Receive FIFO Not Empty Interrupt Line 0: Interrupt assigned to FlexRay interrupt 0 1: Interrupt assigned to FlexRay interrupt 1
4	RXIL	Receive Interrupt Line 0: Interrupt assigned to FlexRay interrupt 0 1: Interrupt assigned to FlexRay interrupt 1
3	TXIL	Transmit Interrupt Line 0: Interrupt assigned to FlexRay interrupt 0 1: Interrupt assigned to FlexRay interrupt 1
2	CYCSL	Cycle Start Interrupt Line 0: Interrupt assigned to FlexRay interrupt 0 1: Interrupt assigned to FlexRay interrupt 1
1	CASL	Collision Avoidance Symbol Interrupt Line 0: Interrupt assigned to FlexRay interrupt 0 1: Interrupt assigned to FlexRay interrupt 1
0	WSTL	Wakeup Status Interrupt Line 0: Interrupt assigned to FlexRay interrupt 0 1: Interrupt assigned to FlexRay interrupt 1

17.2.4.5 FLXA0FREIES — FlexRay Error Interrupt Enable Set Register

The settings in the FlexRay error interrupt enable set register (FLXA0FREIES) and FlexRay error interrupt enable reset register (FLXA0FREIER) determine which status changes in the FlexRay error interrupt register will result in an interrupt.

The enable bits are set by writing to FLXA0FREIES and reset by writing to FLXA0FREIER. Reading from both addresses will result in the same value.

Writing '0' has no effect on the bit value.

Writing a '1' sets the interrupt enable bit.

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: ERAY + 0030_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	TABBE	LTVBE	EDBE	—	—	—	—	—	TABAE	LTVAE	EDAE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	MHFE	IOBAE	IIBAE	EFAE	RFOE	AERRE	CCLE	CCFE	SFOE	SFBME	CNAE	PEMCE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 17.10 FLXA0FREIES Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 27	—	Reserved These bits are always read as 0. When writing, always write 0.
26	TABBE	Transmission Violation Across Boundary Channel B Interrupt Enable 0: Interrupt disabled 1: Interrupt enabled
25	LTVBE	Latest Transmit Violation Channel B Interrupt Enable 0: Interrupt disabled 1: Interrupt enabled
24	EDBE	Error Detected on Channel B Interrupt Enable 0: Interrupt disabled 1: Interrupt enabled
23 to 19	—	Reserved These bits are always read as 0. When writing, always write 0.
18	TABAE	Transmission Violation Across Boundary Channel A Interrupt Enable 0: Interrupt disabled 1: Interrupt enabled
17	LTVAE	Latest Transmit Violation Channel A Interrupt Enable 0: Interrupt disabled 1: Interrupt enabled
16	EDAE	Error Detected on Channel A Interrupt Enable 0: Interrupt disabled 1: Interrupt enabled
15 to 12	—	Reserved These bits are always read as 0. When writing, always write 0.

Table 17.10 FLXA0FREIES Register Contents (2/2)

Bit Position	Bit Name	Function
11	MHFE	Message Handler Constraints Flag Interrupt Enable 0: Interrupt disabled 1: Interrupt enabled
10	IOBAE	Illegal Output Buffer Access Interrupt Enable 0: Interrupt disabled 1: Interrupt enabled
9	IIBAE	Illegal Input Buffer Access Interrupt Enable 0: Interrupt disabled 1: Interrupt enabled
8	EFAE	Empty FIFO Access Interrupt Enable3 0: Interrupt disabled 1: Interrupt enabled
7	RFOE	Receive FIFO Overrun Interrupt Enable 0: Interrupt disabled 1: Interrupt enabled
6	AERRE	Access error interrupt enable 0: Interrupt is disabled. 1: Interrupt is enabled.
5	CCLE	CHI Command Locked Interrupt Enable 0: Interrupt disabled 1: Interrupt enabled
4	CCFE	Clock Correction Failure Interrupt Enable 0: Interrupt disabled 1: Interrupt enabled
3	SFOE	Sync Frame Overflow Interrupt Enable 0: Interrupt disabled 1: Interrupt enabled
2	SFBME	Sync Frames Sync Frame Shortfall Interrupt Enable 0: Interrupt disabled 1: Interrupt enabled
1	CNAE	Command Not Accepted Interrupt Enable 0: Interrupt disabled 1: Interrupt enabled
0	PEMCE	POC Error Mode Changed Interrupt Enable 0: Interrupt disabled 1: Interrupt enabled

17.2.4.6 FLXA0FREIER — FlexRay Error Interrupt Enable Reset Register

The settings in the FlexRay error interrupt enable set register (FLXA0FREIES) and FlexRay error interrupt enable reset register (FLXA0FREIER) determine which status changes in the FlexRay error interrupt register will result in an interrupt.

The enable bits are set by writing to FLXA0FREIES and reset by writing to FLXA0FREIER. Reading from both addresses will result in the same value.

Writing '0' has no effect on the bit value.

Writing a '1' clears the interrupt enable bit.

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: ERAY + 0034_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	TABBD	LTVBD	EDBD	—	—	—	—	—	TABAD	LTVAD	EDAD
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	MHFD	IOBAD	IIBAD	EFAD	RFOD	AERRD	CCLD	CCFD	SFOD	SFBMD	CNAD	PEMCD
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 17.11 FLXA0FREIER Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 27	—	Reserved These bits are always read as 0. When writing, always write 0.
26	TABBD	Transmission Violation Across Boundary Channel B Interrupt Disable 0: Interrupt disabled 1: Interrupt enabled
25	LTVBD	Latest Transmit Violation Channel B Interrupt Disable 0: Interrupt disabled 1: Interrupt enabled
24	EDBD	Error Detected on Channel B Interrupt Disable 0: Interrupt disabled 1: Interrupt enabled
23 to 19	—	Reserved These bits are always read as 0. When writing, always write 0.
18	TABAD	Transmission Violation Across Boundary Channel A Interrupt Disable 0: Interrupt disabled 1: Interrupt enabled
17	LTVAD	Latest Transmit Violation Channel A Interrupt Disable 0: Interrupt disabled 1: Interrupt enabled
16	EDAD	Error Detected on Channel A Interrupt Disable 0: Interrupt disabled 1: Interrupt enabled
15 to 12	—	Reserved These bits are always read as 0. When writing, always write 0.

Table 17.11 FLXA0FREIER Register Contents (2/2)

Bit Position	Bit Name	Function
11	MHFD	Message Handler Constraints Flag Interrupt Disable 0: Interrupt disabled 1: Interrupt enabled
10	IOBAD	Illegal Output Buffer Access Interrupt Disable 0: Interrupt disabled 1: Interrupt enabled
9	IIBAD	Illegal Input Buffer Access Interrupt Disable 0: Interrupt disabled 1: Interrupt enabled
8	EFAD	Empty FIFO Access Interrupt Disable 0: Interrupt disabled 1: Interrupt enabled
7	RFOD	Receive FIFO Overrun Interrupt Disable 0: Interrupt disabled 1: Interrupt enabled
6	AERRD	Access error interrupt disable 0: Interrupt is disabled. 1: Interrupt is enabled.
5	CCLD	CHI Command Locked Interrupt Disable 0: Interrupt disabled 1: Interrupt enabled
4	CCFD	Clock Correction Failure Interrupt Disable 0: Interrupt disabled 1: Interrupt enabled
3	SFOD	Sync Frame Overflow Interrupt Disable 0: Interrupt disabled 1: Interrupt enabled
2	SFBMD	Sync Frames Sync Frame Shortfall Interrupt Disable 0: Interrupt disabled 1: Interrupt enabled
1	CNAD	Command Not Accepted Interrupt Disable 0: Interrupt disabled 1: Interrupt enabled
0	PEMCD	POC Error Mode Changed Interrupt Disable 0: Interrupt disabled 1: Interrupt enabled

17.2.4.7 FLXA0FRSIES — FlexRay Status Interrupt Enable Set Register

The settings in the FlexRay status interrupt enable set register (FLXA0FRSIES) and FlexRay status interrupt enable reset register (FLXA0FRSIER) determine which status changes in the FlexRay status interrupt register will result in an interrupt.

The enable bits are set by writing to FLXA0FRSIES and reset by writing to FLXA0FRSIER. Reading from both addresses will result in the same value.

Writing '0' has no effect on the bit value.

Writing a '1' sets the interrupt enable bit.

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: ERAY + 0038_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	MTSBE	WUPBE	—	—	—	—	—	—	MTSAE	WUPAE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SDSE	MBSIE	SUCSE	SWEE	TOBCE	TIBCE	TI1E	TI0E	NMVCE	RFCLE	RFNEE	RXIE	TXIE	CYCSE	CASE	WSTE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 17.12 FLXA0FRSIES Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 26	—	Reserved These bits are always read as 0. When writing, always write 0.
25	MTSBE	MTS Received on Channel B Interrupt Enable 0: Interrupt disabled 1: Interrupt enabled
24	WUPBE	Wakeup Pattern Channel B Interrupt Enable 0: Interrupt disabled 1: Interrupt enabled
23 to 18	—	Reserved These bits are always read as 0. When writing, always write 0.
17	MTSAE	MTS Received on Channel A Interrupt Enable 0: Interrupt disabled 1: Interrupt enabled
16	WUPAE	Wakeup Pattern Channel A Interrupt Enable 0: Interrupt disabled 1: Interrupt enabled
15	SDSE	Start of Dynamic Segment Interrupt Enable 0: Interrupt disabled 1: Interrupt enabled
14	MBSIE	Message Buffer Status Interrupt Enable 0: Interrupt disabled 1: Interrupt enabled
13	SUCSE	Startup Completed Successfully Interrupt Enable 0: Interrupt disabled 1: Interrupt enabled

Table 17.12 FLXA0FRSIES Register Contents (2/2)

Bit Position	Bit Name	Function
12	SWEE	Stop Watch Event Interrupt Enable 0: Interrupt disabled 1: Interrupt enabled
11	TOBCE	Transfer Output Buffer Completed Interrupt Enable 0: Interrupt disabled 1: Interrupt enabled
10	TIBCE	Transfer Input Buffer Completed Interrupt Enable 0: Interrupt disabled 1: Interrupt enabled
9	TI1E	Timer Interrupt 1 Enable 0: Interrupt disabled 1: Interrupt enabled
8	TI0E	Timer Interrupt 0 Enable 0: Interrupt disabled 1: Interrupt enabled
7	NMVCE	Network Management Vector Changed Interrupt Enable 0: Interrupt disabled 1: Interrupt enabled
6	RFCLE	Receive FIFO Critical Level Interrupt Enable 0: Interrupt disabled 1: Interrupt enabled
5	RFNEE	Receive FIFO Not Empty Interrupt Enable 0: Interrupt disabled 1: Interrupt enabled
4	RXIE	Receive Interrupt Enable 0: Interrupt disabled 1: Interrupt enabled
3	TXIE	Transmit Interrupt Enable 0: Interrupt disabled 1: Interrupt enabled
2	CYCSE	Cycle Start Interrupt Enable 0: Interrupt disabled 1: Interrupt enabled
1	CASE	Collision Avoidance Symbol Interrupt Enable 0: Interrupt disabled 1: Interrupt enabled
0	WSTE	Wakeup Status Interrupt Enable 0: Interrupt disabled 1: Interrupt enabled

17.2.4.8 FLXA0FRSIER — FlexRay Status Interrupt Enable Reset Register

The settings in the FlexRay status interrupt enable set register (FLXA0FRSIES) and FlexRay status interrupt enable reset register (FLXA0FRSIER) determine which status changes in the FlexRay status interrupt register will result in an interrupt.

The enable bits are set by writing to FLXA0FRSIES and reset by writing to FLXA0FRSIER. Reading from both addresses will result in the same value.

Writing '0' has no effect on the bit value.

Writing a '1' clears the interrupt enable bit.

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: ERAY + $003C_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	MTSBD	WUPBD	—	—	—	—	—	—	MTSAD	WUPAD
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SDSD	MBSID	SUCSD	SWED	TOBCD	TIBCD	TI1D	TI0D	NMVCD	RFCLD	RFNED	RXID	TXID	CYCSD	CASD	WSTD
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 17.13 FLXA0FRSIER Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 26	—	Reserved These bits are always read as 0. When writing, always write 0.
25	MTSBD	MTS Received on Channel B Interrupt Disable 0: Interrupt disabled 1: Interrupt enabled
24	WUPBD	Wakeup Pattern Channel B Interrupt Disable 0: Interrupt disabled 1: Interrupt enabled
23 to 18	—	Reserved These bits are always read as 0. When writing, always write 0.
17	MTSAD	MTS Received on Channel A Interrupt Disable 0: Interrupt disabled 1: Interrupt enabled
16	WUPAD	Wakeup Pattern Channel A Interrupt Disable 0: Interrupt disabled 1: Interrupt enabled
15	SDSD	Start of Dynamic Segment Interrupt Disable 0: Interrupt disabled 1: Interrupt enabled
14	MBSID	Message Buffer Status Interrupt Disable 0: Interrupt disabled 1: Interrupt enabled
13	SUCSD	Startup Completed Successfully Interrupt Disable 0: Interrupt disabled 1: Interrupt enabled

Table 17.13 FLXA0FRSIER Register Contents (2/2)

Bit Position	Bit Name	Function
12	SWED	Stop Watch Event Interrupt Disable 0: Interrupt disabled 1: Interrupt enabled
11	TOBCD	Transfer Output Buffer Completed Interrupt Disable 0: Interrupt disabled 1: Interrupt enabled
10	TIBCD	Transfer Input Buffer Completed Interrupt Disable 0: Interrupt disabled 1: Interrupt enabled
9	TI1D	Timer Interrupt 1 Disable 0: Interrupt disabled 1: Interrupt enabled
8	TI0D	Timer Interrupt 0 Disable 0: Interrupt disabled 1: Interrupt enabled
7	NMVCD	Network Management Vector Changed Interrupt Disable 0: Interrupt disabled 1: Interrupt enabled
6	RFCLD	Receive FIFO Critical Level Interrupt Disable 0: Interrupt disabled 1: Interrupt enabled
5	RFNED	Receive FIFO Not Empty Interrupt Disable 0: Interrupt disabled 1: Interrupt enabled
4	RXID	Receive Interrupt Disable 0: Interrupt disabled 1: Interrupt enabled
3	TXID	Transmit Interrupt Disable 0: Interrupt disabled 1: Interrupt enabled
2	CYCSD	Cycle Start Interrupt Disable 0: Interrupt disabled 1: Interrupt enabled
1	CASD	Collision Avoidance Symbol Interrupt Disable 0: Interrupt disabled 1: Interrupt enabled
0	WSTD	Wakeup Status Interrupt Disable 0: Interrupt disabled 1: Interrupt enabled

17.2.4.9 FLXA0FRILE — FlexRay Interrupt Line Enable Register

Each of the two module interrupt lines (FlexRay interrupt 0, FlexRay interrupt 1) can be enabled or disabled separately by programming bit FLXA0FRILE.EINT0 and FLXA0FRILE.EINT1.

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: ERAY + 0040_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	EINT1	EINT0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Table 17.14 FLXA0FRILE Register Contents

Bit Position	Bit Name	Function
31 to 2	—	Reserved These bits are always read as 0. When writing, always write 0.
1	EINT1	Enable FlexRay Interrupt Line 1 0: FlexRay interrupt 1 disabled 1: FlexRay interrupt 1 enabled
0	EINT0	Enable FlexRay Interrupt Line 0 0: FlexRay interrupt 0 disabled 1: FlexRay interrupt 0 enabled

17.2.5 FlexRay Timer Registers

17.2.5.1 FLXA0FRT0C — FlexRay Timer 0 Configuration Register

This register is an absolute timer. It specifies the point in time when a FlexRay timer 0 interrupt occurs as the values of cycle count and macrotick (MT). When the FlexRay timer 0 passes, FLXA0FRSIR.TI0 and FLXA0FROS.T0IS are set to '1'. A timer 0 interrupt then occurs while the FLXA0FROC.T0IE bit is effective.

NOTE

The configuration of timer 0 is compared against the macrotick counter value, there is no separate counter for timer 0.

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: ERAY + 0044_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	T0MO[13:0]													
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	T0CC[6:0]							—	—	—	—	—	—	T0MS	T0RC
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R/W	R/W

Table 17.15 FLXA0FRT0C Register Contents

Bit Position	Bit Name	Function
31, 30	—	Reserved These bits are always read as 0. When writing, always write 0.
29 to 16	T0MO[13:0]	Timer 0 Macrotick Offset Timer 0 macrotick offset
15	—	Reserved This bit is always read as 0. When writing, always write 0.
14 to 8	T0CC[6:0]	Timer 0 Cycle Code timer 0 cycle code
7 to 2	—	Reserved These bits are always read as 0. When writing, always write 0.
1	T0MS	Timer 0 Mode Select 0: Single-shot mode 1: Continuous mode
0	T0RC	Timer 0 Run Control 0: Timer 0 halted 1: Timer 0 running

(1) FLXA0FRT0C.T0MO

Timer 0 Macrotick Offset

Before reconfiguration of the timer, the timer has to be halted first by writing bit FLXA0FRT0C.T0RC to ‘0’.

Configures the macrotick offset from the beginning of the communication cycle where the interrupt is to occur. The FlexRay timer 0 interrupt occurs at this offset for each cycle of the cycle set.

(2) FLXA0FRT0C.T0CC

Timer 0 Cycle Code

Before reconfiguration of the timer, the timer has to be halted first by writing bit FLXA0FRT0C.T0RC to ‘0’.

The 7-bit timer 0 cycle code determines the cycle set used for generation of the FlexRay timer 0 interrupt. For details about the configuration of the cycle code, see **Section 17.3.8.2, Cycle Counter Filtering**.

(3) FLXA0FRT0C.T0MS

Timer 0 Mode Select

Before reconfiguration of the timer, the timer has to be halted first by writing bit FLXA0FRT0C.T0RC to ‘0’.

Configures the timer run mode. In Single-shot mode the timer is deactivated when the timer configuration matches the configured cycle counter and macrotick value.

(4) FLXA0FRT0C.T0RC

Timer 0 Run Control

Timer 0 can be activated (set FLXA0FRT0C.T0RC to ‘1’) when the POC is either in NORMAL_ACTIVE state or in NORMAL_PASSIVE state.

Timer 0 is deactivated when leaving NORMAL_ACTIVE state or NORMAL_PASSIVE state except for transitions between the two states.

17.2.5.2 FLXA0FRT1C — FlexRay Timer 1 Configuration Register

This register is a relative timer. After the specified number of macroticks (MT) has expired, a FlexRay timer 1 interrupt is asserted. When the FlexRay timer 1 passes, FLXA0FRSIR.TI1 and FLXA0FROS.T1IS are set to '1'. A timer 1 interrupt then occurs while the FLXA0FROC.T1IE bit is effective.

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: ERAY + 0048_H

Value after reset: $0002\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	T1MC[13:0]													
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	T1MS	T1RC
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Table 17.16 FLXA0FRT1C Register Contents

Bit Position	Bit Name	Function
31, 30	—	Reserved These bits are always read as 0. When writing, always write 0.
29 to 16	T1MC[13:0]	Timer 1 Macrotick Count Timer 1 macrotick count
15 to 2	—	Reserved These bits are always read as 0. When writing, always write 0.
1	T1MS	Timer 1 Mode Select 0: Single-shot mode 1: Continuous mode
0	T1RC	Timer 1 Run Control 0: Timer 1 halted 1: Timer 1 running

(1) FLXA0FRT1C.T1MC

Timer 1 Macrotick Count

Before reconfiguration of the timer, the timer has to be halted first by writing bit FLXA0FRT1C.T1RC to ‘0’.

Valid values are 2 to 16383 MT in continuous mode

Valid values are 1 to 16383 MT in single-shot mode

When the configured macrotick count is reached the FlexRay timer 1 interrupt is generated.

(2) FLXA0FRT1C.T1MS

Timer 1 Mode Select

Before reconfiguration of the timer, the timer has to be halted first by writing bit FLXA0FRT1C.T1RC to ‘0’.

Configures the timer run mode. In Single-shot mode the timer is deactivated when the timer configuration matches the configured cycle counter and macrotick value.

(3) FLXA0FRT1C.T1RC

Timer 1 Run Control

Timer 1 can be activated (set FLXA0FRT1C.T1RC to ‘1’) as long as the POC is either in NORMAL_ACTIVE state or in NORMAL_PASSIVE state.

Timer 1 is deactivated when leaving NORMAL_ACTIVE state or NORMAL_PASSIVE state except for transitions between the two states.

17.2.5.3 FLXA0FRT2C — FlexRay Timer 2 Configuration Register

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: ERAY + 0844_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	T2MO[13:0]													
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	T2CC[6:0]							—	—	—	—	—	—	T2MS	T2RC
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R/W	R/W

Table 17.17 FLXA0FRT2C Register Contents

Bit Position	Bit Name	Function
31, 30	—	Reserved These bits are always read as 0. When writing, always write 0.
29 to 16	T2MO[13:0]	Timer 2 Macrotick Offset Timer 2 macrotick offset
15	—	Reserved This bit is always read as 0. When writing, always write 0.
14 to 8	T2CC[6:0]	Timer 2 Cycle Code Timer 2 cycle code
7 to 2	—	Reserved These bits are always read as 0. When writing, always write 0.
1	T2MS	Timer 2 Mode Select 0: Single-shot mode 1: Continuous mode
0	T2RC	Timer 2 Run Control 0: Timer halted 1: Timer running

(1) FLXA0FRT2C.T2MO

Timer 2 Macrotick Offset

Write 0 to the FLXA0FRT2C.T2RC bit and halt the timer before changing the value of the timer.

Configures the macrotick offset from the beginning of the communication cycle where the timer 2 interrupt is generated. The timer 2 interrupt is generated at this offset for each cycle of the cycle set.

(2) FLXA0FRT2C.T2CC

Timer 2 Cycle Code

Before reconfiguration of the timer, the timer has to be halted first by writing 0 to bit FLXA0FRT2C.T2RC.

Configures the cycle set which generates timer 2 interrupt with the 7-bit timer 2 cycle code. For details, see **Section 17.3.8.2, Cycle Counter Filtering**.

(3) FLXA0FRT2C.T2MS

Timer 2 Mode Select

Before reconfiguration of the timer, the timer has to be halted first by writing 0 to bit FLXA0FRT2C.T2RC.

Configures the timer run mode. In single-shot mode, the timer is deactivated when the timer configuration matches the configured cycle counter and macrotick value.

(4) FLXA0FRT2C.T2RC

Timer 2 Run Control

Timer 2 can be operated when the POC is in NORMAL_ACTIVE or NORMAL_PASSIVE state. (set FLXA0FRT2C.T2RC to ‘1’)

Timer 2 is deactivated when leaving NORMAL_ACTIVE state or NORMAL_PASSIVE state except for transitions between the two states.

17.2.5.4 FLXA0FRSTPW1 — FlexRay Stop Watch Register 1

The stop watch is activated by the following trigger events.

- FlexRay interrupt 0 or FlexRay interrupt 1
- Writing bit FLXA0FRSTPW1.SSWT to '1'

With the macrotick counter increment following next to the stop watch activation the actual cycle counter and macrotick values are captured in register FLXA0FRSTPW1 while the slot counter values for channel A and B are captured in register FLXA0FRSTPW2.

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: ERAY + $004C_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	SMTV[13:0]													
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	SCCV[5:0]						—	EINT1	EINT0	—	SSWT	EDGE	SWMS	ESWT
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R	R/W	R/W	R/W	R/W

Table 17.18 FLXA0FRSTPW1 Register Contents (1/2)

Bit Position	Bit Name	Function
31, 30	—	Reserved These bits are always read as 0. When writing, always write 0.
29 to 16	SMTV[13:0]	Stop Watch Captured Macrotick Value Stop watch captured macrotick value
15, 14	—	Reserved These bits are always read as 0. When writing, always write 0.
13 to 8	SCCV[5:0]	Stop Watch Captured Cycle Counter Value Stop watch captured cycle counter value
7	—	Reserved This bit is always read as 0. When writing, always write 0.
6	EINT1	Enable FlexRay Interrupt 1 Trigger 0: Stop watch trigger by FlexRay interrupt 1 disabled 1: FlexRay interrupt 1 event triggers stop watch
5	EINT0	Enable FlexRay Interrupt 0 Trigger 0: Stop watch trigger by FlexRay interrupt 0 disabled 1: FlexRay interrupt 0 event triggers stop watch
4	—	Reserved This bit is always read as 0. When writing, always write 0.
3	SSWT	Software Stop Watch Trigger 0: Software trigger reset 1: Stop watch activated by software trigger
2	EDGE	Stop Watch Trigger Edge Select 0: Falling edge 1: Rising edge

Table 17.18 FLXA0FRSTPW1 Register Contents (2/2)

Bit Position	Bit Name	Function
1	SWMS	Stop Watch Mode Select 0: Single-shot mode 1: Continuous mode
0	ESWT	Enable Hardware Stop Watch Trigger 0: Stop watch trigger disabled 1: Stop watch trigger enabled

(1) FLXA0FRSTPW1.SMTV

Stop Watch Captured Macrotick Value

State of the macrotick counter when the stop watch event occurred.

(2) FLXA0FRSTPW1.SCCV

Stop Watch Captured Cycle Counter Value

State of the cycle counter when the stop watch event occurred.

(3) FLXA0FRSTPW1.EINT1

Enable FlexRay Interrupt 1 Trigger

Enables stop watch trigger by FlexRay interrupt 1 when FLXA0FRSTPW1.ESWT = ‘1’.

(4) FLXA0FRSTPW1.EINT0

Enable FlexRay Interrupt 0 Trigger

Enables stop watch trigger by FlexRay interrupt 0 when FLXA0FRSTPW1.ESWT = ‘1’.

(5) FLXA0FRSTPW1.SSWT

Software Stop Watch Trigger

Bits FLXA0FRSTPW1.ESWT and FLXA0FRSTPW1.SSWT cannot be set to ‘1’ simultaneously. In this case the write access to the register is ignored, and both bits keep their previous values. Either the external stop watch trigger or the software stop watch trigger may be used.

Writing 1 in this bit activates the stop watch. This bit is reset to ‘0’ after the cycle count and slot count, and macrotick (MT) value are stored in the FlexRay stop watch register.

(6) FLXA0FRSTPW1.EDGE

Stop Watch Trigger Edge Select

(7) FLXA0FRSTPW1.SWMS

Stop Watch Mode Select

(8) FLXA0FRSTPW1.ESWT

Enable Stop Watch Trigger

Bits FLXA0FRSTPW1.ESWT and FLXA0FRSTPW1.SSWT cannot be set to ‘1’ simultaneously. In this case the write access to the register is ignored, and both bits keep their previous values. Either the external stop watch trigger or the software stop watch trigger may be used.

If enabled, a FlexRay interrupt 0 event or a FlexRay interrupt 1 event activates the stop watch.

In single-shot mode, this bit is reset to ‘0’ after the cycle count and slot count, and macrotick (MT) value are stored in the FlexRay stop watch register.

17.2.5.5 FLXA0FRSTPW2 — FlexRay Stop Watch Register 2

Access: This register can be read in 8-, 16-, or 32-bit units.

Address: ERAY + 0050_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	SSCVB[10:0]										
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SSCVA[10:0]										
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 17.19 FLXA0FRSTPW2 Register Contents

Bit Position	Bit Name	Function
31 to 27	—	Reserved These bits are always read as 0. When writing, always write 0.
26 to 16	SSCVB[10:0]	Stop Watch Captured Slot Counter Value Channel B
15 to 11	—	Reserved These bits are always read as 0. When writing, always write 0.
10 to 0	SSCVA[10:0]	Stop Watch Captured Slot Counter Value Channel A

(1) FLXA0FRSTPW2.SSCVB

Stop Watch Captured Slot Counter Value Channel B

State of the slot counter for channel B when the stop watch event occurred.

(2) FLXA0FRSTPW2.SSCVA

Stop Watch Captured Slot Counter Value Channel A

State of the slot counter for channel A when the stop watch event occurred.

17.2.6 CC Control Registers

This section describes the registers provided by the CC to allow the host to control the operation of the CC. The FlexRay protocol specification requires the host to write application configuration data in CONFIG state only. Please consider that the configuration registers are not locked for writing in DEFAULT_CONFIG state.

The configuration data is reset when DEFAULT_CONFIG state is entered from reset. To change POC state from DEFAULT_CONFIG to CONFIG state the host has to apply CHI command CONFIG. If the host wants the CC to leave CONFIG state, the host has to execute the lock release sequence as described in **Section 17.2.3.1, FLXA0FRLCK — FlexRay Lock Register**.

17.2.6.1 FLXA0FRSUCC1 — FlexRay SUC Configuration Register 1

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: ERAY + 0080_H

Value after reset: $0C40\ 1080_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	CCHB	CCHA	MTSB	MTSA	HCSE	TSM	WUCS	PTA[4:0]				
Value after reset	0	0	0	0	1	1	0	0	0	1	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CSA[4:0]					—	TXSY	TXST	PBSY	—	—	—	CMD[3:0]			
Value after reset	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 17.20 FLXA0FRSUCC1 Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 28	—	Reserved These bits are always read as 0. When writing, always write 0.
27	CCHB	Connected to Channel B Configures pChannels 0: Node not connected to channel B 1: Node connected to channel B (default after reset)
26	CCHA	Connected to Channel A Configures pChannels 0: Node not connected to channel A 1: Node connected to channel A (default after reset)
25	MTSB	Select Channel B for MTS Transmission 0: Channel B disabled for MTS transmission 1: Channel B selected for MTS transmission
24	MTSA	Select Channel A for MTS Transmission 0: Channel A disabled for MTS transmission 1: Channel A selected for MTS transmission
23	HCSE	Halt due to Clock Sync Error Configures pAllowHaltDueToClock 0: CC will enter or remain in NORMAL_PASSIVE 1: CC will enter HALT state
22	TSM	Transmission Slot Mode Configures pSingleSlotEnabled 0: ALL Slot Mode 1: SINGLE Slot Mode (default after hard reset)

Table 17.20 FLXA0FRSUCC1 Register Contents (2/2)

Bit Position	Bit Name	Function
21	WUCS	Wakeup Channel Select Configures pWakeupChannel 0: Send wakeup pattern on channel A 1: Send wakeup pattern on channel B
20 to 16	PTA[4:0]	Passive to Active Configure pAllowPassiveToActive
15 to 11	CSA[4:0]	Cold Start Attempts Configure gColdStartAttempts
10	—	Reserved This bit is always read as 0. When writing, always write 0.
9	TXSY	Transmit Sync Frame in Key Slot Configure pKeySlotUsedForSync 0: No sync frame transmission in key slot, node is neither sync nor coldstart node 1: Key slot used to transmit sync frame, node is sync node
8	TXST	Transmit Startup Frame in Key Slot Configure pKeySlotUsedForStartup 0: No startup frame transmission in key slot, node is non-coldstarter 1: Key slot used to transmit startup frame, node is leading or following coldstarter
7	PBSY	POC Busy Flag 0: POC not busy, FLXA0FRSUCC1.CMD writeable 1: POC is busy, FLXA0FRSUCC1.CMD locked
6 to 4	—	Reserved These bits are always read as 0. When writing, always write 0.
3 to 0	CMD[3:0]	CHI Command Vector 0000: command_not_accepted 0001: CONFIG 0010: READY 0011: WAKEUP 0100: RUN 0101: ALL_SLOTS 0110: HALT 0111: FREEZE 1000: SEND_MTS 1001: ALLOW_COLDSTART 1010: RESET_STATUS_INDICATORS 1100: CLEAR_RAMS others: reserved

(1) FLXA0FRSUCC1.CCHB

Connected to Channel B

A user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

Configures whether the node is connected to channel B (pChannels).

(2) FLXA0FRSUCC1.CCHA

Connected to Channel A

A user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

Configures whether the node is connected to channel A (pChannels).

(3) FLXA0FRSUCC1.MTSB

Select Channel B for MTS Transmission

A user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

FLXA0FRSUCC1.MTSB may also be changed outside DEFAULT_CONFIG or CONFIG state when the write to FLXA0FRSUCC1 register is directly preceded by the unlock sequence for the Configuration Lock Key as described in **Section 17.2.3.1, FLXA0FRLCK — FlexRay Lock Register**. This may be combined with CHI command SEND_MTS. If both bits FLXA0FRSUCC1.MTSA and FLXA0FRSUCC1.MTSB are set to '1' an MTS symbol will be transmitted on both channels when requested by writing FLXA0FRSUCC1.CMD = "1000".

The bit selects channel B for MTS symbol transmission.

(4) FLXA0FRSUCC1.MTSA

Select Channel A for MTS Transmission

A user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

FLXA0FRSUCC1.MTSA may also be changed outside DEFAULT_CONFIG or CONFIG state when the write to FLXA0FRSUCC1 register is directly preceded by the unlock sequence for the Configuration Lock Key as described in **Section 17.2.3.1, FLXA0FRLCK — FlexRay Lock Register**. This may be combined with CHI command SEND_MTS. If both bits FLXA0FRSUCC1.MTSA and FLXA0FRSUCC1.MTSB are set to '1' an MTS symbol will be transmitted on both channels when requested by writing FLXA0FRSUCC1.CMD = "1000".

The bit selects channel A for MTS symbol transmission.

(5) FLXA0FRSUCC1.HCSE

Halt due to Clock Sync Error

A user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

Controls the transition to HALT state due to a clock synchronization error (pAllowHaltDueToClock).

(6) FLXA0FRSUCC1.TSM

Transmission Slot Mode

A user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

Selects the initial transmission slot mode (pSingleSlotEnabled).

In SINGLE slot mode the CC may only transmit in the preconfigured key slot. The key slot ID is configured in the header section of message buffer 0 respectively message buffers 0 and 1 depending on bit FLXA0FRMRC.SPLM.

In case FLXA0FRSUCC1.TSM = '1', message buffer 0 respectively message buffers 0,1 can be (re)configured in DEFAULT_CONFIG or CONFIG state only. In ALL slot mode the CC may transmit in all slots.

FLXA0FRSUCC1.TSM is a configuration bit which can only be set or reset by the host.

The CC changes to ALL slot mode when the host successfully applied the ALL_SLOTS command by writing FLXA0FRSUCC1.CMD = "0101" in POC states NORMAL_ACTIVE or NORMAL_PASSIVE. The actual slot mode is monitored by FLXA0FRCCSV.SLM.

(7) FLXA0FRSUCC1.WUCS

Wakeup Channel Select

The user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

With this bit the host selects the channel on which the CC sends the Wakeup pattern (pWakeupChannel).

(8) FLXA0FRSUCC1.PTA

Passive to Active Condition Setting

A user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

Valid values are 0 to 31 even/odd cycle pairs.

Defines the number of consecutive even/odd cycle pairs that must have valid clock correction terms before the CC is allowed to transit from NORMAL_PASSIVE to NORMAL_ACTIVE state (pAllowPassiveToActive).

If set to “00000” the CC is not allowed to transit from NORMAL_PASSIVE to NORMAL_ACTIVE state.

(9) FLXA0FRSUCC1.CSA

Cold Start Attempt Count Setting

A user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

Must be identical in all nodes of a cluster.

Valid values are 2 to 31.

Configures the maximum number of attempts that a cold starting node is permitted to try to start up the network without receiving any valid response from another node (gColdStartAttempts).

(10) FLXA0FRSUCC1.TXSY

Transmit Sync Frame in Key Slot

A user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

Defines whether the key slot is used to transmit sync frames (pKeySlotUsedForSync).

NOTE

The protocol requires that both bits FLXA0FRSUCC1.TXST and FLXA0FRSUCC1.TXSY are set for coldstart nodes.

(11) FLXA0FRSUCC1.TXST

Transmit Startup Frame in Key Slot

A user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

Defines whether the key slot is used to transmit startup frames (pKeySlotUsedForStartup).

NOTE

The protocol requires that both bits FLXA0FRSUCC1.TXST and FLXA0FRSUCC1.TXSY are set for coldstart nodes.

(12) FLXA0FRSUCC1.PBSY

POC Busy

Signals that the POC is busy and cannot accept a command from the host. FLXA0FRSUCC1.CMD is locked against write accesses.

Set to '1' after reset during initialization of internal RAM blocks.

(13) FLXA0FRSUCC1.CMD

CHI Command Vector

The host may write any CHI command at any time, but certain commands are enabled only in certain POC states. If a command is not enabled, it will not be executed, the CHI command vector FLXA0FRSUCC1.CMD will be reset to "0000" = command_not_accepted, and flag FLXA0FREIR.CNA will be set to '1'.

In general the host must check FLXA0FRSUCC1.PBSY before writing a new CHI command.

In case the previous CHI command has not yet completed, FLXA0FREIR.CCL is set to '1' together with FLXA0FREIR.CNA; the CHI command needs to be repeated.

Except for HALT state, a POC state change command applied while the CC is already in the requested POC state neither causes a state change nor will FLXA0FREIR.CNA be set.

Reading FLXA0FRSUCC1.CMD shows whether the last CHI command was accepted. The actual POC state is monitored by FLXA0FRCCSV.POCS.

- command_not_accepted

  FLXA0FRSUCC1.CMD is reset to "0000" due to one of the following conditions:

  – Illegal command applied by the host
  – Host applied command to leave CONFIG state without preceding config lock key
  – Host applied new command while execution of the previous host command has not completed
  – Host writes command_not_accepted

When FLXA0FRSUCC1.CMD is reset to "0000", FLXA0FREIR.CNA is set to '1', and enabled, an interrupt is generated.

Commands which are not accepted are not executed.

CONFIG

Go to POC state CONFIG when called in POC states DEFAULT_CONFIG, or READY. When called in HALT state the CC transits to POC state DEFAULT_CONFIG. When called in any other state, FLXA0FRSUCC1.CMD will be reset to "0000" = command_not_accepted.

READY

Go to POC state READY when called in POC states CONFIG, NORMAL_ACTIVE, NORMAL_PASSIVE, STARTUP, or WAKEUP. When called in any other state, FLXA0FRSUCC1.CMD will be reset to “0000” = command_not_accepted.

WAKEUP

Go to POC state WAKEUP when called in POC state READY. When called in any other state, FLXA0FRSUCC1.CMD will be reset to “0000” = command_not_accepted.

RUN

Go to POC state STARTUP when called in POC state READY. When called in any other state, FLXA0FRSUCC1.CMD will be reset to “0000” = command_not_accepted.

ALL_SLOTS

Leave SINGLE slot mode and go to ALL-SLOTS mode after successful startup/integration at the next end of cycle when called in POC states NORMAL_ACTIVE or NORMAL_PASSIVE. When called in any other state, FLXA0FRSUCC1.CMD will be reset to “0000” = command_not_accepted.

HALT

Set halt request FLXA0FRCCSV.HRQ to ‘1’ and go to POC state HALT at the next end of cycle when called in POC states NORMAL_ACTIVE or NORMAL_PASSIVE. When called in any other state, FLXA0FRSUCC1.CMD will be reset to “0000” = command_not_accepted.

FREEZE

Set the freeze status indicator FLXA0FRCCSV.FSI to ‘1’ and go to POC state HALT immediately. Can be called from any state.

SEND_MTS

Send single MTS symbol during the next following symbol window on the channel configured by FLXA0FRSUCC1.MTSA, FLXA0FRSUCC1.MTSB, when called in POC state NORMAL_ACTIVE after CC entered ALL slot mode (FLXA0FRCCSV.SLM = “11”). When called in any other state, or when called while a previously requested MTS has not yet been transmitted, FLXA0FRSUCC1.CMD will be reset to “0000” = command_not_accepted.

ALLOW_COLDSTART

The command resets FLXA0FRCCSV.CSI to enable the node to become leading coldstarter. When called in states DEFAULT_CONFIG, CONFIG, or HALT, FLXA0FRSUCC1.CMD will be reset to “0000” = command_not_accepted. To become leading coldstarter it is also required that both FLXA0FRSUCC1.TXST and FLXA0FRSUCC1.TXSY are set.

RESET_STATUS_INDICATORS

Resets status flags FLXA0FRCCSV.FSI, FLXA0FRCCSV.HRQ, FLXA0FRCCSV.CSNI, and FLXA0FRCCSV.CSAI to their default values. Can be called in POC states READY and STARTUP. When called in any other state, FLXA0FRSUCC1.CMD will be reset to “0000” = command_not_accepted.

CLEAR_RAMS

Sets FLXA0FRMHDS.CRAM to '1' when called in DEFAULT_CONFIG or CONFIG state. When called in any other state, FLXA0FRSUCC1.CMD will be reset to "0000" = command_not_accepted.

FLXA0FRMHDS.CRAM is also set to '1' when the CC leaves reset. By setting FLXA0FRMHDS.CRAM all internal RAM blocks are initialized to zero. During the initialization of the RAMs, FLXA0FRSUCC1.PBSY will show POC busy. Access to the configuration and status registers is possible during execution of CHI command CLEAR_RAMS.

The initialization of the internal RAM block requires 2048 bus clock cycles. There should be no host access to input buffer or output buffer during initialization of the internal RAM blocks after reset or after assertion of CHI command CLEAR_RAMS.

Before asserting CHI command CLEAR_RAMS, confirm that no transfer between message RAM and input buffer or output buffer or the temporary buffer RAMs is ongoing and that the data transfer handler is disabled (FLXA0FRITS.ITS = 0 and FLXA0FROTS.OTS = 0). This command also resets the message buffer status registers FLXA0FRMHDS, FLXA0FRLDTS, FLXA0FRFSR, FLXA0FRMHDF, FLXA0FRTXRQ1/2/3/4, FLXA0FRNDAT1/2/3/4, and FLXA0FRMBSC1/2/3/4.

NOTES

1. All accepted commands with exception of CLEAR_RAMS and SEND_MTS will cause a change of the POC state in the FlexRay domain after at most 8 cycles of the slower of the two clocks "bus clock" and "FlexRay", assumed that POC was not busy when the command was applied and that no POC state change was forced by bus activity in that time frame. Reading register FLXA0FRCCSV will show data that is additionally delayed by synchronization from the FlexRay domain to the bus clock domain. The maximum additional delay is 12 cycles of the slower of the two clocks "bus clock" and "FlexRay".
2. When a terminal restarts transfer as a leading coldstart node after transfer was stopped by using the FREEZE or READY command, the startup frame may not be transmitted in cycle 0. This depends on the internal state of the FlexRay module. This phenomenon occurs when the startup frame is set in a slot with a number from 1 to 7.
   This phenomenon does not occur in a coldstart after a hardware reset.
   Even if this phenomenon does occur, a second try of the coldstart will succeed.
   While the coldstart time will be prolonged, this phenomenon does not otherwise affect a coldstart of the FlexRay system.
   To avoid this phenomenon, allocate the Startup/Sync frame to static slot 8 or a slot with a higher number.

Table 17.21 below references the CHI commands from the FlexRay Protocol Specification (Section 2.2.1.1, Table 2.2) to the FlexRay CHI command vector FLXA0FRSUCC1.CMD.

Table 17.21 Reference to CHI Host Command Summary from FlexRay Protocol Specification

CHI Command	Where Processed (POC States)	CHI Command Vector CMD
ALL_SLOTS	POC: normal active, POC: normal passive	ALL_SLOTS
ALLOW_COLDSTART	All except POC: default config, POC: config, POC: halt	ALLOW_COLDSTART
CONFIG	POC: default config, POC: ready	CONFIG
CONFIG_COMPLETE	POC: config	Unlock sequence & READY
DEFAULT_CONFIG	POC: halt	CONFIG
FREEZE	All	FREEZE
HALT	POC: normal active, POC: normal passive	HALT
READY	All except POC: default config, POC: config, POC: ready, POC: halt	READY
RUN	POC: ready	RUN
WAKEUP	POC: ready	WAKEUP

17.2.6.2 FLXA0FRSUCC2 — FlexRay SUC Configuration Register 2

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: ERAY + 0084_H

Value after reset: $0100\ 0504_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	LTN[3:0]				—	—	—	LT[20:16]				
Value after reset	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LT[15:0]															
Value after reset	0	0	0	0	0	1	0	1	0	0	0	0	0	1	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 17.22 FLXA0FRSUCC2 Register Contents

Bit Position	Bit Name	Function
31 to 28	—	Reserved These bits are always read as 0. When writing, always write 0.
27 to 24	LTN[3:0]	Listen Timeout Noise Configure (gListenNoise - 1)
23 to 21	—	Reserved These bits are always read as 0. When writing, always write 0.
20 to 0	LT[20:0]	Listen Timeout Configure pdListenTimeout

(1) FLXA0FRSUCC2.LTN

Listen Timeout Noise

A user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

Valid values are 2 to 16.

FLXA0FRSUCC2.LTN must be configured identical in all nodes of a cluster.

Configures the upper limit for startup and wakeup listen timeout in the presence of noise expressed as a multiple of pdListenTimeout.

NOTE

The wakeup/startup noise timeout is calculated as follows:

pdListenTimeout × gListenNoise = FLXA0FRSUCC2.LT × (FLXA0FRSUCC2.LTN + 1)

(2) FLXA0FRSUCC2.LT

Listen Timeout

A user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

Valid values are 1284 to 1283846 µT.

Configures wakeup/startup listen timeout in µT.

17.2.6.3 FLXA0FRSUCC3 — FlexRay SUC Configuration Register 3

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: ERAY + 0088_H

Value after reset: $0000\ 0011_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	WCF[3:0]				WCP[3:0]			
Value after reset	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 17.23 FLXA0FRSUCC3 Register Contents

Bit Position	Bit Name	Function
31 to 8	—	Reserved These bits are always read as 0. When writing, always write 0.
7 to 4	WCF[3:0]	Maximum Without Clock Correction Fatal (condition for transition to HALT state) Configure gMaxWithoutClockCorrectionFatal
3 to 0	WCP[3:0]	Maximum Without Clock Correction Passive (condition for transition to NORMAL_PASSIVE state) Configure gMaxWithoutClockCorrectionPassive

(1) FLXA0FRSUCC3.WCF

Maximum Without Clock Correction Fatal (condition for transition to HALT state)

A user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

Valid values are 1 to 15 cycle pairs.

Setting must be identical in all nodes of a cluster.

Defines the number of consecutive even/odd cycle pairs with missing clock correction terms that will cause a transition from NORMAL_ACTIVE or NORMAL_PASSIVE to HALT state.

NOTE

The transition to HALT state is prevented if FLXA0FRSUCC1.HCSE is not set.

(2) FLXA0FRSUCC3.WCP

Maximum Without Clock Correction Passive (transition to NORMAL_PASSIVE state)

A user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

Valid values are 1 to 15 cycle pairs.

Setting must be identical in all nodes of a cluster.

Defines the number of consecutive even/odd cycle pairs with missing clock correction terms that will cause a transition from NORMAL_ACTIVE to NORMAL_PASSIVE state.

17.2.6.4 FLXA0FRNEMC — FlexRay NEM Configuration Register

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: ERAY + $008C_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	NML[3:0]			
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Table 17.24 FLXA0FRNEMC Register Contents

Bit Position	Bit Name	Function
31 to 4	—	Reserved These bits are always read as 0. When writing, always write 0.
3 to 0	NML[3:0]	Network Management Vector Length Configure gNetworkManagementVectorLength

(1) FLXA0FRNEMC.NML

Network Management Vector Length

A user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

Valid values are 0 to 12 bytes.

The configured length must be identical in all nodes of a cluster.

These bits configure the length of the NM vector.

17.2.6.5 FLXA0FRPRTC1 — FlexRay PRT Configuration Register 1

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: ERAY + 0090_H

Value after reset: $084C\ 0633_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RWP[5:0]						—	RXW[8:0]								
Value after reset	0	0	0	0	1	0	0	0	0	1	0	0	1	1	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BRP[1:0]		SPP[1:0]		—	CASM[6:0]							TSST[3:0]			
Value after reset	0	0	0	0	0	1	1	0	0	0	1	1	0	0	1	1
R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 17.25 FLXA0FRPRTC1 Register Contents

Bit Position	Bit Name	Function
31 to 26	RWP[5:0]	Repetitions of Tx Wakeup Pattern Configure pWakeupPattern
25	—	Reserved This bit is always read as 0. When writing, always write 0.
24 to 16	RXW[8:0]	Wakeup Symbol Receive Window Length Configure gdWakeupSymbolRxWindow
15, 14	BRP[1:0]	Baud Rate Prescaler Configure gdSampleClockPeriod and pSamplesPerMicrotick 00 = 10 Mbps 01 = 5 Mbps 10 = 2.5 Mbps 11 = 2.5 Mbps
13, 12	SPP[1:0]	Strobe Point Position Configure Strobe point position 00 = Sample 5 01 = Sample 4 10 = Sample 6 11 = Sample 5
11	—	Reserved This bit is always read as 0. When writing, always write 0.
10 to 4	CASM[6:0]	Collision Avoidance Symbol Max Configure gdCASRxLowMax
3 to 0	TSST[3:0]	Transmission Start Sequence Transmitter Configure gdTSSTransmitter

(1) FLXA0FRPRTC1.RWP

Repetitions of Tx Wakeup Pattern

A user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

Valid values are 2 to 63.

Configures the number of repetitions (sequences) of the Tx wakeup symbol.

(2) FLXA0FRPRTC1.RXW

Wakeup Symbol Receive Window Length

A user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

Valid values are 76 to 301 bit times.

Must be identical in all nodes of a cluster.

Configures the number of bit times used by the node to test the duration of the received wakeup pattern.

(3) FLXA0FRPRTC1.BRP

Baud Rate Prescaler

A user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

The Baud Rate Prescaler configures the baud rate on the FlexRay bus. The baud rates listed below are valid with a sample clock set to 80 MHz. One bit time always consists of 8 samples independent of the configured baud rate.

00 = 10 MBit/s

gdSampleClockPeriod = 12.5 ns = 1 * ‘sample clock’

pSamplesPerMicrotick = 2 (1 µT = 25 ns)

01 = 5 MBit/s

gdSampleClockPeriod = 25 ns = 2 * ‘sample clock’

pSamplesPerMicrotick = 1 (1 µT = 25 ns)

10, 11 = 2.5 MBit/s

gdSampleClockPeriod = 50 ns = 4 * ‘sample clock’

pSamplesPerMicrotick = 1 (1 µT = 50 ns)

(4) FLXA0FRPRTC1.SPP

Strobe Point Position

A user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

Defines the sample count value for strobing. The strobed bit value is set to the voted value when the sample count is incremented to the value configured by FLXA0FRPRTC1.SPP.

NOTE

The current revision 2.1 of the FlexRay protocol requires that FLXA0FRPRTC1.SPP = “00”. The alternate strobe point positions could be used to compensate for asymmetries in the physical layer.

(5) FLXA0FRPRTC1.CASM

Collision Avoidance Symbol Max

A user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

CASM6 is fixed to '1'.

Valid values are 67 to 99 bit times.

Configures the upper limit of the acceptance window for a collision avoidance symbol (CAS).

(6) FLXA0FRPRTC1.TSST

Transmission Start Sequence Transmitter

A user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

Valid values are 3 to 15 bit times.

Must be identical in all nodes of a cluster.

Configures the duration of the Transmission Start Sequence (TSS) in terms of bit times (1 bit time = 4 µT = 100ns at 10 Mbps).

17.2.6.6 FLXA0FRPRTC2 — FlexRay PRT Configuration Register 2

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: ERAY + 0094_H

Value after reset: $0F2D\ 0A0E_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	TXL[5:0]						TXI[7:0]							
Value after reset	0	0	0	0	1	1	1	1	0	0	1	0	1	1	0	1
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	RXL[5:0]						—	—	RXI[5:0]					
Value after reset	0	0	0	0	1	0	1	0	0	0	0	0	1	1	1	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Table 17.26 FLXA0FRPRTC2 Register Contents

Bit Position	Bit Name	Function
31, 30	—	Reserved These bits are always read as 0. When writing, always write 0.
29 to 24	TXL[5:0]	Wakeup Symbol Transmit Low Configure gdWakeupSymbolTxLow
23 to 16	TXI[7:0]	Wakeup Symbol Transmit Idle Configure gdWakeupSymbolTxIdle
15, 14	—	Reserved These bits are always read as 0. When writing, always write 0.
13 to 8	RXL[5:0]	Wakeup Symbol Receive Low Configure gdWakeupSymbolRxLow
7, 6	—	Reserved These bits are always read as 0. When writing, always write 0.
5 to 0	RXI[5:0]	Wakeup Symbol Rx Idle Configure gdWakeupSymbolRxIdle

(1) FLXA0FRPRTC2.TXL

Wakeup Symbol Transmit Low

A user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

Valid values are 15 to 60 bit times.

Must be identical in all nodes of a cluster.

Configures the number of bit times used by the node to transmit the low phase of the wakeup symbol.

(2) FLXA0FRPRTC2.TXI

Wakeup Symbol Transmit Idle

A user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

Valid values are 45 to 180 bit times.

Must be identical in all nodes of a cluster.

Configures the number of bit times used by the node to transmit the idle phase of the wakeup symbol.

(3) FLXA0FRPRTC2.RXL

Wakeup Symbol Receive Low

A user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

Valid values are 10 to 55 bit times.

Must be identical in all nodes of a cluster.

Configures the number of bit times used by the node to test the duration of the low phase of the received wakeup symbol.

(4) FLXA0FRPRTC2.RXI

Wakeup Symbol Rx Idle

A user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

Valid values are 14 to 59 bit times.

Must be identical in all nodes of a cluster.

Configures the number of bit times used by the node to test the duration of the idle phase of the received wakeup symbol.

17.2.6.7 FLXA0FRMHDC — FlexRay MHD Configuration Register

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: ERAY + 0098_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	SLT[12:0]												
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	SFDL[6:0]						
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 17.27 FLXA0FRMHDC Register Contents

Bit Position	Bit Name	Function
31 to 29	—	Reserved These bits are always read as 0. When writing, always write 0.
28 to 16	SLT[12:0]	Start of Latest Transmit Configure pLatestTx
15 to 7	—	Reserved These bits are always read as 0. When writing, always write 0.
6 to 0	SFDL[6:0]	Static Frame Data Length Configure gPayloadLengthStatic

(1) FLXA0FRMHDC.SLT

Start of Latest Transmit

A user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

Valid values are 0 to 7981 minislots.

Configures the maximum minislot value allowed before inhibiting frame transmission in the dynamic segment of the cycle. There is no transmission in dynamic segment if FLXA0FRMHDC.SLT is set to zero.

(2) FLXA0FRMHDC.SFDL

Static Frame Data Length

A user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

Valid values are 0 to 127.

The payload length must be identical in all nodes of a cluster.

Configures the cluster-wide payload length for all frames sent in the static segment in double bytes.

17.2.6.8 FLXA0FRGTUC1 — FlexRay GTU Configuration Register 1

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: ERAY + $00A0_H$

Value after reset: $0000\ 0280_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	UT[19:16]			
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	UT[15:0]															
Value after reset	0	0	0	0	0	0	1	0	1	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 17.28 FLXA0FRGTUC1 Register Contents

Bit Position	Bit Name	Function
31 to 20	—	Reserved These bits are always read as 0. When writing, always write 0.
19 to 0	UT[19:0]	Microtick per Cycle Configure pMicroPerCycle

(1) FLXA0FRGTUC1.UT

Microtick per Cycle

A user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

Valid values are 640 to 640000 µT.

Configures the duration of the communication cycle in microticks.

17.2.6.9 FLXA0FRGTUC2 — FlexRay GTU Configuration Register 2

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: ERAY + $00A4_H$

Value after reset: $0002\ 000A_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	SNM[3:0]			
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	MPC[13:0]													
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 17.29 FLXA0FRGTUC2 Register Contents

Bit Position	Bit Name	Function
31 to 20	—	Reserved These bits are always read as 0. When writing, always write 0.
19 to 16	SNM[3:0]	Sync Node Max
15, 14	—	Reserved These bits are always read as 0. When writing, always write 0.
13 to 0	MPC[13:0]	Macrotick Per Cycle Configure gMacroPerCycle

(1) FLXA0FRGTUC2.SNM

Sync Node Max

A user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

Valid values are 2 to 15.

Must be identical in all nodes of a cluster.

Maximum number of frames within a cluster with sync frame indicator bit SYN set to ‘1’.

(2) FLXA0FRGTUC2.MPC

Macrotick Per Cycle

A user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

Valid values are 10 to 16000 MT.

The cycle length must be identical in all nodes of a cluster.

Configures the duration of one communication cycle in macroticks.

17.2.6.10 FLXA0FRGTUC3 — FlexRay GTU Configuration Register 3

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: ERAY + $00A8_H$

Value after reset: $0202\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	MIOB[6:0]							—	MIOA[6:0]						
Value after reset	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	UIOB[7:0]								UIOA[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 17.30 FLXA0FRGTUC3 Register Contents

Bit Position	Bit Name	Function
31	—	Reserved This bit is always read as 0. When writing, always write 0.
30 to 24	MIOB[6:0]	Macrotick Initial Offset Channel B Configure pMacroInitialOffset[B]
23	—	Reserved This bit is always read as 0. When writing, always write 0.
22 to 16	MIOA[6:0]	Macrotick Initial Offset Channel A Configure pMacroInitialOffset[A]
15 to 8	UIOB[7:0]	Microtick Initial Offset Channel B Configure pMicroInitialOffset[B]
7 to 0	UIOA[7:0]	Microtick Initial Offset Channel A Configure pMicroInitialOffset[A]

(1) FLXA0FRGTUC3.MIOB

Macrotick Initial Offset Channel B

A user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

Valid values are 2 to 72 MT.

Must be identical in all nodes of a cluster.

Configures the number of macroticks between the static slot boundary and the subsequent macrotick boundary of the secondary time reference point based on the nominal macrotick duration.

(2) FLXA0FRGTUC3.MIOA

Macrotick Initial Offset Channel A

A user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

Valid values are 2 to 72 MT.

Must be identical in all nodes of a cluster.

Configures the number of macroticks between the static slot boundary and the subsequent macrotick boundary of the secondary time reference point based on the nominal macrotick duration.

(3) FLXA0FRGTUC3.UIOB

Microtick Initial Offset Channel B

A user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

Valid values are 0 to 240 µT.

Configures the number of microticks between the actual time reference point on channel B and the subsequent macrotick boundary of the secondary time reference point. The parameter depends on pDelayCompensation [B] and therefore has to be set for each channel independently.

(4) FLXA0FRGTUC3.UIOA

Microtick Initial Offset Channel A

A user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

Valid values are 0 to 240 µT.

Configures the number of microticks between the actual time reference point on channel A and the subsequent macrotick boundary of the secondary time reference point. The parameter depends on pDelayCompensation [A] in the FLXA0FRGTUC5 register and therefore has to be set for each channel independently.

17.2.6.11 FLXA0FRGTUC4 — FlexRay GTU Configuration Register 4

For details about configuration of FLXA0FRGTUC4.NIT and FLXA0FRGTUC4.OCS, see **Section 17.3.2.5, Configuration of NIT Start and Offset Correction Start**.

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: ERAY + $00AC_H$

Value after reset: $0008\ 0007_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	OCS[13:0]													
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	NIT[13:0]													
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 17.31 FLXA0FRGTUC4 Register Contents

Bit Position	Bit Name	Function
31, 30	—	Reserved These bits are always read as 0. When writing, always write 0.
29 to 16	OCS[13:0]	Offset Correction Start Configure (gOffsetCorrectionStart - 1)
15, 14	—	Reserved These bits are always read as 0. When writing, always write 0.
13 to 0	NIT[13:0]	Network Idle Time Start Configure (gMacroPerCycle -gdNIT - 1)

(1) FLXA0FRGTUC4.OCS

Offset Correction Start

A user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

Valid values are 8 to 15998 MT.

For cluster consisting of E-Ray implementations only, it is sufficient to program FLXA0FRGTUC4.OCS = FLXA0FRGTUC4.NIT + 1.

Must be identical in all nodes of a cluster.

Determines the start of the offset correction within the NIT phase, calculated from start of cycle.

(2) FLXA0FRGTUC4.NIT

Network Idle Time Start

A user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

Valid values are 7 to 15997 MT.

Must be identical in all nodes of a cluster.

Configures the starting point of the Network Idle Time NIT at the end of the communication cycle expressed in terms of macroticks from the beginning of the cycle. The start of NIT is recognized if Macrotick = gMacroPerCycle -gdNIT -1 and the increment pulse of Macrotick is set.

17.2.6.12 FLXA0FRGTUC5 — FlexRay GTU Configuration Register 5

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: ERAY + $00B0_H$

Value after reset: $0E00\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DEC[7:0]								—	—	—	CDD[4:0]				
Value after reset	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DCB[7:0]								DCA[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 17.32 FLXA0FRGTUC5 Register Contents

Bit Position	Bit Name	Function
31 to 24	DEC[7:0]	Decoding Correction Configure pDecodingCorrection
23 to 21	—	Reserved These bits are always read as 0. When writing, always write 0.
20 to 16	CDD[4:0]	Cluster Drift Damping Configure pClusterDriftDamping
15 to 8	DCB[7:0]	Delay Compensation Channel B Configure pDelayCompensation[B]
7 to 0	DCA[7:0]	Delay Compensation Channel A Configure pDelayCompensation[A]

(1) FLXA0FRGTUC5.DEC

Decoding Correction

A user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

Valid values are 14 to 143 µT.

Configures the decoding correction value in microticks used to determine the primary time reference point.

(2) FLXA0FRGTUC5.CDD

Cluster Drift Damping

A user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

Valid values are 0 to 20 µT.

Configures the cluster drift damping value in microticks used in clock synchronization to minimize accumulation of rounding errors.

(3) FLXA0FRGTUC5.DCB

Delay Compensation Channel B

A user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

Valid values are 0 to 200 µT.

Used to compensate for reception delays on channel B. This covers assumed propagation delay up to cPropagationDelayMax for microticks in the range of 0.0125 to 0.05µs. In practice, the minimum of the propagation delays of all sync nodes should be applied.

(4) FLXA0FRGTUC5.DCA

Delay Compensation Channel A

A user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

Valid values are 0 to 200 µT.

Used to compensate for reception delays on channel A. This covers assumed propagation delay up to cPropagationDelayMax for microticks in the range of 0.0125 to 0.05µs. In practice, the minimum of the propagation delays of all sync nodes should be applied.

17.2.6.13 FLXA0FRGTUC6 — FlexRay GTU Configuration Register 6

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: ERAY + $00B4_H$

Value after reset: $0002\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	MOD[10:0]										
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
R/W	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	ASR[10:0]										
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 17.33 FLXA0FRGTUC6 Register Contents

Bit Position	Bit Name	Function
31 to 27	—	Reserved These bits are always read as 0. When writing, always write 0.
26 to 16	MOD[10:0]	Maximum Oscillator Drift Configure pdMaxDrift
15 to 11	—	Reserved These bits are always read as 0. When writing, always write 0.
10 to 0	ASR[10:0]	Accepted Startup Range Configure pdAcceptedStartupRange

(1) FLXA0FRGTUC6.MOD

Maximum Oscillator Drift

A user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

Valid values are 2 to 1923 µT.

Maximum drift offset between two nodes that operate with unsynchronized clocks over one communication cycle in µT.

(2) FLXA0FRGTUC6.ASR

Accepted Startup Range

A user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

Valid values are 0 to 1875 µT.

Number of microticks constituting the expanded range of measured deviation for startup frames during integration.

17.2.6.14 FLXA0FRGTUC7 — FlexRay GTU Configuration Register 7

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: ERAY + $00B8_H$

Value after reset: $0002\ 0004_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	NSS[9:0]									
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	SSL[9:0]									
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 17.34 FLXA0FRGTUC7 Register Contents

Bit Position	Bit Name	Function
31 to 26	—	Reserved These bits are always read as 0. When writing, always write 0.
25 to 16	NSS[9:0]	Number of Static Slots Configure gNumberOfStaticSlots
15 to 10	—	Reserved These bits are always read as 0. When writing, always write 0.
9 to 0	SSL[9:0]	Static Slot Length Configure gdStaticSlot

(1) FLXA0FRGTUC7.NSS

Number of Static Slots

A user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

Valid values are 2 to 1023.

The number of static slots must be identical in all nodes of a cluster.

Configures the number of static slots in a cycle.

(2) FLXA0FRGTUC7.SSL

Static Slot Length

A user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

Valid values are 4 to 659 MT.

The static slot length must be identical in all nodes of a cluster.

Configures the length of a static slot in macroticks.

17.2.6.15 FLXA0FRGTUC8 — FlexRay GTU Configuration Register 8

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: ERAY + $00BC_H$

Value after reset: $0000\ 0002_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	NMS[12:0]												
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	MSL[5:0]					
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
R/W	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Table 17.35 FLXA0FRGTUC8 Register Contents

Bit Position	Bit Name	Function
31 to 29	—	Reserved These bits are always read as 0. When writing, always write 0.
28 to 16	NMS[12:0]	Number of Minislots Configure gNumberOfMinislots
15 to 6	—	Reserved These bits are always read as 0. When writing, always write 0.
5 to 0	MSL[5:0]	Minislot Length Configure gdMinislot

(1) FLXA0FRGTUC8.NMS

Number of Minislots

A user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

Valid values are 0 to 7986.

The number of minislots must be identical in all nodes of a cluster.

Configures the number of minislots within the dynamic segment of a cycle.

(2) FLXA0FRGTUC8.MSL

Minislot Length

A user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

Valid values are 2 to 63 MT.

The minislot length must be identical in all nodes of a cluster.

Configures the length of a minislot in macroticks.

17.2.6.16 FLXA0FRGTUC9 — FlexRay GTU Configuration Register 9

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: ERAY + $00C0_H$

Value after reset: $0000\ 0101_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DSI[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	MAPO[4:0]					—	—	APO[5:0]					
Value after reset	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Table 17.36 FLXA0FRGTUC9 Register Contents

Bit Position	Bit Name	Function
31 to 18	—	Reserved These bits are always read as 0. When writing, always write 0.
17, 16	DSI[1:0]	Dynamic Slot Idle Phase Configure gdDynamicSlotIdlePhase
15 to 13	—	Reserved These bits are always read as 0. When writing, always write 0.
12 to 8	MAPO[4:0]	Minislot Action Point Offset Configure gdMinislotActionPointOffset
7, 6	—	Reserved These bits are always read as 0. When writing, always write 0.
5 to 0	APO[5:0]	Action Point Offset Configure gdActionPointOffset

(1) FLXA0FRGTUC9.DSI

Dynamic Slot Idle Phase

A user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

Valid values are 0 to 2 Minislots.

Must be identical in all nodes of a cluster.

Configures the duration of the dynamic slot idle phase in the number of minislots. The duration has to be greater or equal than the idle detection time.

(2) FLXA0FRGTUC9.MAPO

Minislot Action Point Offset

A user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

Valid values are 1 to 31 MT.

Must be identical in all nodes of a cluster.

Configures the action point offset in macroticks within the minislots of the dynamic segment.

(3) FLXA0FRGTUC9.APO

Action Point Offset

A user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

Valid values are 1 to 63 MT.

Must be identical in all nodes of a cluster.

Configures the action point offset in macroticks within static slots and symbol window.

17.2.6.17 FLXA0FRGTUC10 — FlexRay GTU Configuration Register 10

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: ERAY + $00C4_H$

Value after reset: $0002\ 0005_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	MRC[10:0]										
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
R/W	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	MOC[13:0]													
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 17.37 FLXA0FRGTUC10 Register Contents

Bit Position	Bit Name	Function
31 to 27	—	Reserved These bits are always read as 0. When writing, always write 0.
26 to 16	MRC[10:0]	Maximum Rate Correction Configure pRateCorrectionOut
15, 14	—	Reserved These bits are always read as 0. When writing, always write 0.
13 to 0	MOC[13:0]	Maximum Offset Correction Configure pOffsetCorrectionOut

(1) FLXA0FRGTUC10.MRC

Maximum Rate Correction

A user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

Valid values are 2 to 1923 µT.

Holds the maximum permitted rate correction value to be applied by the internal clock synchronization algorithm. The CC checks only the internal rate correction value against the maximum rate correction value (absolute value).

(2) FLXA0FRGTUC10.MOC

Maximum Offset Correction

A user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

Valid values are 5 to 15266 µT.

Holds the maximum permitted offset correction value (absolute value) to be applied by the internal clock synchronization algorithm (absolute value). The CC checks only the internal offset correction value against the maximum offset correction value.

17.2.6.18 FLXA0FRGTUC11 — FlexRay GTU Configuration Register 11

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: ERAY + $00C8_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	ERC[2:0]			—	—	—	—	—	EOC[2:0]		
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	ERCC[1:0]		—	—	—	—	—	—	EOCC[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R/W	R/W

Table 17.38 FLXA0FRGTUC11 Register Contents

Bit Position	Bit Name	Function
31 to 27	—	Reserved These bits are always read as 0. When writing, always write 0.
26 to 24	ERC[2:0]	External Rate Correction Configure pExternRateCorrection
23 to 19	—	Reserved These bits are always read as 0. When writing, always write 0.
18 to 16	EOC[2:0]	External Offset Correction Configure pExternOffsetCorrection
15 to 10	—	Reserved These bits are always read as 0. When writing, always write 0.
9, 8	ERCC[1:0]	External Rate Correction Control Configure vExternRateControl 00: External rate correction is prohibited. 01: External rate correction is prohibited. 10: Subtract 11: Add
7 to 2	—	Reserved These bits are always read as 0. When writing, always write 0.
1, 0	EOCC[1:0]	External Offset Correction Control Configure vExternOffsetControl 00: External offset correction is prohibited. 01: External offset correction is prohibited. 10: Subtract 11: Add

(1) FLXA0FRGTUC11.ERC

External Rate Correction

A user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

Valid values are 0 to 7 µT.

Holds the external rate correction value in microticks to be applied by the internal clock synchronization algorithm. The value is subtracted or added from or to the calculated rate correction value. The value is applied during NIT.

(2) FLXA0FRGTUC11.EOC

External Offset Correction

A user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

Valid values are 0 to 7 µT.

Holds the external offset correction value in microticks to be applied by the internal clock synchronization algorithm. The value is subtracted or added from or to the calculated offset correction value. The value is applied during NIT.

(3) FLXA0FRGTUC11.ERCC

External Rate Correction Control

Should be modified only outside NIT (Network Idle Time).

By writing to FLXA0FRGTUC11.ERCC the external rate correction is enabled as specified below.

00 = External rate correction is prohibited.

01 = External rate correction is prohibited.

10 = Subtract

External rate correction value subtracted from calculated rate correction value

11 = Add

External rate correction value added to calculated rate correction value

(4) FLXA0FRGTUC11.EOCC

External Offset Correction Control

Should be modified only outside NIT (Network Idle Time).

By writing to FLXA0FRGTUC11.EOCC the external offset correction is enabled as specified below.

00 = External offset correction is prohibited.

01 = External offset correction is prohibited.

10 = Subtract

External offset correction value subtracted from calculated offset correction value

11 = Add

External offset correction value added to calculated offset correction value

17.2.7 CC Status Registers

During 8/16-bit accesses to status variables coded with more than 8/16-bit, the variable might be updated by the CC between two accesses (non-atomic read accesses).

17.2.7.1 FLXA0FRCCSV — FlexRay CC Status Vector Register

Access: This register can be read in 8-, 16-, or 32-bit units.

Address: ERAY + 0100_H

Value after reset: $0010\ 4000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	PSL[5:0]						RCA[4:0]					WSV[2:0]		
Value after reset	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	CSI	CSAI	CSNI	—	—	SLM[1:0]		HRQ	FSI	POCS[5:0]					
Value after reset	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 17.39 FLXA0FRCCSV Register Contents

Bit	Symbol	Function
31, 30	—	Reserved These bits are always read as 0. When writing, always write 0.
29 to 24	PSL[5:0]	POC Status Log Flags Status of FLXA0FRCCSV.POCS immediately before entering HALT state.
23 to 19	RCA[4:0]	Remaining Coldstart Attempts Flags Indicate vRemainingColdstartAttempts
18 to 16	WSV[2:0]	Wakeup Status Flags Indicate vPOC!WakeupStatus
15	—	Reserved This bit is always read as 0. When writing, always write 0.
14	CSI	Cold Start Inhibit Flag Indicates vColdStartInhibit 0: Cold starting of node enabled 1: Cold starting of node disabled
13	CSAI	Coldstart Abort Indicator Flag
12	CSNI	Coldstart Noise Indicator Flag Indicates vPOC!ColdstartNoise
11, 10	—	Reserved These bits are always read as 0. When writing, always write 0.
9, 8	SLM[1:0]	Slot Mode Flags Indicate vPOC!SlotMode 00: SINGLE 01: reserved 10: ALL_PENDING 11: ALL
7	HRQ	Halt Request Flag Indicates vPOC!CHIHaltRequest
6	FSI	Freeze Status Indicator Flag Indicates vPOC!Freeze
5 to 0	POCS[5:0]	Protocol Operation Control Status Flags

(1) FLXA0FRCCSV.PSL

POC Status Log Flags

Set the value of FLXA0FRCCSV.POCS immediately before entering HALT state.

Set to HALT when FREEZE command is applied during HALT state and FLXA0FRCCSV.FSI is not already set i.e. the HALT state was not reached by FREEZE command.

Reset to "000000_B" when leaving HALT state.

(2) FLXA0FRCCSV.RCA

Remaining Coldstart Attempts Flags

Indicates the number of remaining coldstart attempts (vRemainingColdstartAttempts).

The initial value of FLXA0FRCCSV.RCA during CONFIG and DEFAULT_CONFIG state is also FLXA0FRSUCC1.CSA.

The RUN command resets this counter to the maximum number of coldstart attempts as configured by FLXA0FRSUCC1.CSA.

(3) FLXA0FRCCSV.WSV

Wakeup Status Flags

Indicates the status of the current wakeup attempt (vPOC!WakeupStatus).

Reset to “0” when entering Wakeup state, by CHI command RESET_STATUS_INDICATORS, or by transition from DEFAULT_CONFIG to CONFIG state

000 = UNDEFINED

Wakeup not yet executed by the CC.

001 = RECEIVED_HEADER

Set when the CC finishes wakeup due to the reception of a frame header without coding violation on either channel in WAKEUP_LISTEN state.

010 = RECEIVED_WUP

Set when the CC finishes wakeup due to the reception of a valid wakeup pattern on the configured wakeup channel in WAKEUP_LISTEN state.

011 = COLLISION_HEADER

Set when the CC stops wakeup due to a detected collision during wakeup pattern transmission by receiving a valid header on either channel.

100 = COLLISION_WUP

Set when the CC stops wakeup due to a detected collision during wakeup pattern transmission by receiving a valid wakeup pattern on the configured wakeup channel.

101 = COLLISION_UNKNOWN

Set when the CC stops wakeup by leaving WAKEUP_DETECT state after expiration of the wakeup timer without receiving a valid wakeup pattern or a valid frame header.

110 = TRANSMITTED

Set when the CC has successfully completed the transmission of the wakeup pattern.

111 = reserved

(4) FLXA0FRCCSV.CSI

Cold Start Inhibit Flag

Indicates that the node is disabled from cold starting (vColdStartInhibit).

The flag is set to “1” whenever the POC enters READY state due to CHI command READY.

The flag has to be reset under control of the host by CHI command ALLOW_COLDSTART (FLXA0FRSUCC1.CMD = “1001”).

(5) FLXA0FRCCSV.CSAI

Coldstart Abort Indicator Flag

Coldstart aborted.

Reset by CHI command RESET_STATUS_INDICATORS or by transition from HALT to DEFAULT_CONFIG state or from READY to STARTUP state.

(6) FLXA0FRCCSV.CSNI

Coldstart Noise Indicator Flag

Indicates that the cold start procedure occurred under noisy conditions (vPOC!ColdstartNoise).

Reset by CHI command RESET_STATUS_INDICATORS or by transition from HALT to DEFAULT_CONFIG state or from READY to STARTUP state.

(7) FLXA0FRCCSV.SLM

Slot Mode Flags

Indicates the actual slot mode of the POC (vPOC!SlotMode) in states READY, WAKEUP, STARTUP, NORMAL_ACTIVE, and NORMAL_PASSIVE.

Default value is SINGLE. Changes to ALL, depending on FLXA0FRSUCC1.TSM.

In NORMAL_ACTIVE or NORMAL_PASSIVE state the CHI command ALL_SLOTS will change the slot mode from SINGLE over ALL_PENDING to ALL.

Set FLXA0FRSUCC1.TSM to SINGLE except for NORMAL_ACTIVE or NORMAL_PASSIVE.

(8) FLXA0FRCCSV.HRQ

Halt RequestFlag

Indicates that a request from the host has been received to halt the POC at the end of the communication cycle (vPOC!CHIHaltRequest).

Reset by transition from HALT to DEFAULT_CONFIG state or when entering READY state.

(9) FLXA0FRCCSV.FSI

Freeze Status Indicator Flag

Indicates that the POC has entered the HALT state due to CHI command FREEZE or due to an error condition requiring an immediate POC halt (vPOC!Freeze).

Reset by transition from HALT to DEFAULT_CONFIG state.

(10) FLXA0FRCCSV.POCS

Protocol Operation Control Status Flag

Indicates the actual state of operation of the CC Protocol Operation Control

00 0000 = DEFAULT_CONFIG state

00 0001 = READY state

00 0010 = NORMAL_ACTIVE state

00 0011 = NORMAL_PASSIVE state

00 0100 = HALT state

00 0110…00 1110 = reserved

00 1111 = CONFIG state

Indicates the actual state of operation of the POC in the wakeup path

01 0000 = WAKEUP_STANDBY state

01 0001 = WAKEUP_LISTEN state

01 0010 = WAKEUP_SEND state

01 0011 = WAKEUP_DETECT state

Indicates the actual state of operation of the POC in the startup path

10 0000 = STARTUP_PREPARE state

10 0001 = COLDSTART_LISTEN state

10 0010 = COLDSTART_COLLISION_RESOLUTION state

10 0011 = COLDSTART_CONSISTENCY_CHECK state

10 0100 = COLDSTART_GAP state

10 0101 = COLDSTART_JOIN State

10 0110 = INTEGRATION_COLDSTART_CHECK state

10 0111 = INTEGRATION_LISTEN state

10 1000 = INTEGRATION_CONSISTENCY_CHECK state

10 1001 = INITIALIZE_SCHEDULE state

10 1010 = ABORT_STARTUP state

10 1011 = STARTUP_SUCCESS state

Others = reserved

17.2.7.2 FLXA0FRCCEV — FlexRay CC Error Vector Register

Access: This register can be read in 8-, 16-, or 32-bit units.

Address: ERAY + 0104_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	PTAC[4:0]					ERRM[1:0]		—	—	CCFC[3:0]			
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 17.40 FLXA0FRCCEV Register Contents

Bit Position	Bit Name	Function
31 to 13	—	Reserved These bits are always read as 0. When writing, always write 0.
12 to 8	PTAC[4:0]	Passive to Active Count Indicate vAllowPassiveToActive
7, 6	ERRM[1:0]	Error Mode Flags Indicates vPOC!ErrorMode 00: ACTIVE 01: PASSIVE 10: COMM_HALT 11: reserved
5, 4	—	Reserved These bits are always read as 0. When writing, always write 0.
3 to 0	CCFC[3:0]	Clock Correction Failed Counter Indicate vClockCorrectionFailed

(1) FLXA0FRCCEV.PTAC

Passive to Active Count

Indicates the number of consecutive even/odd cycle pairs that have passed with valid rate and offset correction terms, while the node is waiting to transit from NORMAL_PASSIVE state to NORMAL_ACTIVE state. The transition takes place when FLXA0FRCCEV.PTAC equals FLXA0FRSUCC1.PTA -1.

Reset by transition from HALT to DEFAULT_CONFIG state or when entering READY state.

(2) FLXA0FRCCEV.ERRM

Error Mode Flags

Indicates the actual error mode of the POC (vPOC!ErrorMode).

Reset by transition from HALT to DEFAULT_CONFIG state or when entering READY state.

(3) FLXA0FRCCEV.CCFC

Clock Correction Failed Counter

Indicates the clock correction failed counter of the POC (vClockCorrectionFailed).

The Clock Correction Failed Counter is incremented by one at the end of any odd communication cycle where either the missing offset correction error or missing rate correction error are active.

The Clock Correction Failed Counter is reset to '0' at the end of an odd communication cycle if neither the offset correction failed nor the rate correction failed errors are active.

The Clock Correction Failed Counter stops at 15.

Reset by transition from HALT to DEFAULT_CONFIG state or when entering READY state.

17.2.7.3 FLXA0FRSCV — FlexRay Slot Counter Value Register

Access: This register can be read in 8-, 16-, or 32-bit units.

Address: ERAY + 0110_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	SCCB[10:0]										
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	SCCA[10:0]										
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 17.41 FLXA0FRSCV Register Contents

Bit Position	Bit Name	Function
31 to 27	—	Reserved These bits are always read as 0. When writing, always write 0.
26 to 16	SCCB[10:0]	Slot Counter Channel B Indicates vSlotCounter[B]
15 to 11	—	Reserved These bits are always read as 0. When writing, always write 0.
10 to 0	SCCA[10:0]	Slot Counter Channel A Indicates vSlotCounter[A]

(1) FLXA0FRSCV.SCCB

Slot Counter Channel B

Current slot counter value on channel B (vSlotCounter[B]). The value is incremented by the CC and reset at the start of a communication cycle.

Reset when leaving CONFIG state or when entering STARTUP state.

(2) FLXA0FRSCV.SCCA

Slot Counter Channel A

Current slot counter value on channel A (vSlotCounter[A]). The value is incremented by the CC and reset at the start of a communication cycle.

Reset when leaving CONFIG state or when entering STARTUP state.

17.2.7.4 FLXA0FRMTCCV — FlexRay Macrotick and Cycle Counter Value Register

Access: This register can be read in 8-, 16-, or 32-bit units.

Address: ERAY + 0114_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	CCV[5:0]					
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	MTV[13:0]													
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 17.42 FLXA0FRMTCCV Register Contents

Bit Position	Bit Name	Function
31 to 22	—	Reserved These bits are always read as 0. When writing, always write 0.
21 to 16	CCV[5:0]	Cycle Counter Value Indicates vCycleCounter
15, 14	—	Reserved These bits are always read as 0. When writing, always write 0.
13 to 0	MTV[13:0]	Macrotick Value Indicates vMacrotick

(1) FLXA0FRMTCCV.CCV

Cycle Counter Value

Current cycle counter value (vCycleCounter). The value is incremented by the CC at the start of a communication cycle.

Reset when leaving CONFIG state or when entering STARTUP state.

(2) FLXA0FRMTCCV.MTV

Macrotick Value

Current macrotick value (vMacrotick). The value is incremented by the CC and reset at the start of a communication cycle.

Reset when leaving CONFIG state or when entering STARTUP state.

17.2.7.5 FLXA0FRRCV — FlexRay Rate Correction Value Register

Access: This register can be read in 8-, 16-, or 32-bit units.

Address: ERAY + 0118_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	RCV[11:0]											
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 17.43 FLXA0FRRCV Register Contents

Bit Position	Bit Name	Function
31 to 12	—	Reserved These bits are always read as 0. When writing, always write 0.
11 to 0	RCV[11:0]	Rate Correction Value Flags Indicates vRateCorrection

(1) FLXA0FRRCV.RCV

Rate Correction Value Flags

Indicates internal rate correction value (vRateCorrection/ two's complement) before limitation. If the FLXA0FRRCV.RCV value exceeds the limits defined by FLXA0FRGTUC10.MRC, flag FLXA0FRSFS.RCLR is set to ‘1’.

Reset when leaving CONFIG state or when entering STARTUP state.

NOTE

The external rate correction value is added to the limited rate correction value.

17.2.7.6 FLXA0FROCV — FlexRay Offset Correction Value Register

Access: This register can be read in 8-, 16-, or 32-bit units.

Address: ERAY + $011C_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	OCV[18:16]		
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OCV[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 17.44 FLXA0FROCV Register Contents

Bit Position	Bit Name	Function
31 to 19	—	Reserved These bits are always read as 0. When writing, always write 0.
18 to 0	OCV[18:0]	Offset Correction Value Flags Indicate vOffsetCorrection

(1) FLXA0FROCV.OCV

Offset Correction Value Flags

Indicate offset correction value (vOffsetCorrection/ two's complement) before limitation. If the FLXA0FROCV.OCV value exceeds the limits defined by FLXA0FRGTUC10.MOC, flag FLXA0FRSFS.OCLR is set to ‘1’.

Reset when leaving CONFIG state or when entering STARTUP state.

NOTE

The external offset correction value is added to the limited offset correction value.

17.2.7.7 FLXA0FRSFS — FlexRay Sync Frame Status Register

Access: This register can be read in 8-, 16-, or 32-bit units.

Address: ERAY + 0120_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	RCLR	MRCS	OCLR	MOCS
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	VSBO[3:0]				VSBE[3:0]				VSAO[3:0]				VSAE[3:0]			
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 17.45 FLXA0FRSFS Register Contents

Bit Position	Bit Name	Function
31 to 20	—	Reserved These bits are always read as 0. When writing, always write 0.
19	RCLR	Rate Correction Limit Reached Flag 0: Rate correction below limit 1: Rate correction limit reached
18	MRCS	Missing Rate Correction Signal Flag 0: Rate correction signal valid 1: Missing rate correction signal
17	OCLR	Offset Correction Limit Reached Flag 0: Offset correction below limit 1: Offset correction limit reached
16	MOCS	Missing Offset Correction Signal Flag 0: Offset correction signal valid 1: Missing offset correction signal
15 to 12	VSBO[3:0]	Valid Sync Frames Channel B, odd communication cycle
11 to 8	VSBE[3:0]	Valid Sync Frames Channel B, even communication cycle
7 to 4	VSAO[3:0]	Valid Sync Frames Channel A, odd communication cycle
3 to 0	VSAE[3:0]	Valid Sync Frames Channel A, even communication cycle

(1) FLXA0FRSFS.RCLR

Rate Correction Limit Reached Flag

The rate correction limit reached flag signals to the host, that the rate correction value has exceeded its limit as defined by FLXA0FRGTUC10.MRC10 - MRC0. The flag is updated by the CC at start of offset correction phase.

Reset when leaving CONFIG state or when entering STARTUP state, INTEGRATION_COLDSTART_CHECK state or INTEGRATION_CONSISTENCY_CHECK state.

(2) FLXA0FRSFS.MRCS

Missing Rate Correction Signal Flag

The missing rate correction flag signals to the host, that no rate correction calculation can be performed because no pairs of even/odd sync frames were received. The flag is updated by the CC at start of offset correction phase.

Reset when leaving CONFIG state or when entering STARTUP state, INTEGRATION_COLDSTART_CHECK state or INTEGRATION_CONSISTENCY_CHECK state.

(3) FLXA0FRSFS.OCLR

Offset Correction Limit Reached Flag

The offset correction limit reached flag signals to the host, that the offset correction value has exceeded its limit as defined by FLXA0FRGTUC10.MOC. The flag is updated by the CC at start of offset correction phase.

Reset when leaving CONFIG state or when entering STARTUP state, INTEGRATION_COLDSTART_CHECK state or INTEGRATION_CONSISTENCY_CHECK state.

(4) FLXA0FRSFS.MOCS

Missing Offset Correction Signal Flag

The missing offset correction flag signals to the host, that no offset correction calculation can be performed because no sync frames were received. The flag is updated by the CC at start of offset correction phase.

Reset when leaving CONFIG state or when entering STARTUP state, INTEGRATION_COLDSTART_CHECK state or INTEGRATION_CONSISTENCY_CHECK state.

(5) FLXA0FRSFS.VSBO

Valid Sync Frames Channel B, odd communication cycle

These bits are only valid when FLXA0FRSUCC1.CCHB is ‘1’.

Holds the number of valid sync frames received on channel B in the odd communication cycle. If transmission of sync frames is enabled by FLXA0FRSUCC1.TXSY the value is incremented by one. The value is updated during the NIT of each odd communication cycle.

Reset when leaving CONFIG state or when entering STARTUP state, INTEGRATION_COLDSTART_CHECK state or INTEGRATION_CONSISTENCY_CHECK state.

(6) FLXA0FRSFS.VSBE

Valid Sync Frames Channel B, even communication cycle

These bits are only valid when FLXA0FRSUCC1.CCHB is ‘1’.

Holds the number of valid sync frames received on channel B in the even communication cycle. If transmission of sync frames is enabled by FLXA0FRSUCC1.TXSY the value is incremented by one. The value is updated during the NIT of each even communication cycle.

Reset when leaving CONFIG state or when entering STARTUP state, INTEGRATION_COLDSTART_CHECK state or INTEGRATION_CONSISTENCY_CHECK state.

(7) FLXA0FRSFS.VSAO

Valid Sync Frames Channel A, odd communication cycle

These bits are only valid when FLXA0FRSUCC1.CCHA is ‘1’.

Holds the number of valid sync frames received on channel A in the odd communication cycle. If transmission of sync frames is enabled by FLXA0FRSUCC1.TXSY the value is incremented by one. The value is updated during the NIT of each odd communication cycle.

Reset when leaving CONFIG state or when entering STARTUP state, INTEGRATION_COLDSTART_CHECK state or INTEGRATION_CONSISTENCY_CHECK state.

(8) FLXA0FRSFS.VSAE

Valid Sync Frames Channel A, even communication cycle

These bits are only valid when FLXA0FRSUCC1.CCHA is ‘1’.

Holds the number of valid sync frames received on channel A in the even communication cycle. If transmission of sync frames is enabled by FLXA0FRSUCC1.TXSY the value is incremented by one. The value is updated during the NIT of each even communication cycle.

Reset when leaving CONFIG state or when entering STARTUP state, INTEGRATION_COLDSTART_CHECK state or INTEGRATION_CONSISTENCY_CHECK state.

17.2.7.8 FLXA0FRSWNIT — FlexRay Symbol Window and NIT Status Register

Symbol window related status information is updated by the CC at the end of the symbol window for each channel. NIT related status information is updated by the CC at the end of the NIT for each channel.

During startup the status data is not updated.

Access: This register can be read in 8-, 16-, or 32-bit units.

Address: ERAY + 0124_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	SBNB	SENB	SBNA	SENA	MTSB	MTSA	TCSB	SBSB	SESB	TCSA	SBSA	SESA
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 17.46 FLXA0FRSWNIT Register Contents (1/2)

Bit Position	Symbol	Function
31 to 12	—	Reserved These bits are always read as 0. When writing, always write 0.
11	SBNB	Slot Boundary Violation during NIT Channel B Flag 0: No slot boundary violation detected 1: Slot boundary violation during NIT detected on channel B
10	SENB	Syntax Error during NIT Channel B Flag 0: No syntax error detected 1: Syntax error during NIT detected on channel B
9	SBNA	Slot Boundary Violation during NIT Channel A Flag 0: No slot boundary violation detected 1: Slot boundary violation during NIT detected on channel A
8	SENA	Syntax Error during NIT Channel A Flag 0: No syntax error detected 1: Syntax error during NIT detected on channel A
7	MTSB	MTS Received on Channel B Flag 0: No MTS symbol received on channel B 1: MTS symbol received on channel B
6	MTSA	MTS Received on Channel A Flag 0: No MTS symbol received on channel A 1: MTS symbol received on channel A
5	TCSB	Transmission Conflict in Symbol Window Channel B Flag 0: No transmission conflict detected 1: Transmission conflict in symbol window detected on channel B
4	SBSB	Slot Boundary Violation in Symbol Window Channel B Flag 0: No slot boundary violation detected 1: Slot boundary violation during symbol window detected on channel B
3	SESB	Syntax Error in Symbol Window Channel B Flag 0: No syntax error detected 1: Syntax error during symbol window detected on channel B

Table 17.46 FLXA0FRSWNIT Register Contents (2/2)

Bit Position	Symbol	Function
2	TCSA	Transmission Conflict in Symbol Window Channel A Flag 0: No transmission conflict detected 1: Transmission conflict in symbol window detected on channel A
1	SBSA	Slot Boundary Violation in Symbol Window Channel A Flag 0: No slot boundary violation detected 1: Slot boundary violation during symbol window detected on channel A
0	SESA	Syntax Error in Symbol Window Channel A Flag 0: No syntax error detected 1: Syntax error during symbol window detected on channel A

(1) FLXA0FRSWNIT.SBNB

Indicates a Slot Boundary Violation during NIT Channel B Flag (vSS!BViolationB).

Reset when leaving CONFIG state or when entering STARTUP state.

(2) FLXA0FRSWNIT.SENB

Indicates a Syntax Error during NIT Channel B Flag (vSS!SyntaxErrorB).

Reset when leaving CONFIG state or when entering STARTUP state.

(3) FLXA0FRSWNIT.SBNA

Indicates a Slot Boundary Violation during NIT Channel A Flag (vSS!BViolationA).

Reset when leaving CONFIG state or when entering STARTUP state.

(4) FLXA0FRSWNIT.SENA

Indicates a Syntax Error during NIT Channel A Flag (vSS!SyntaxErrorA).

Reset when leaving CONFIG state or when entering STARTUP state.

(5) FLXA0FRSWNIT.MTSB

Indicates a MTS Received on Channel B Flag (vSS!ValidMTSB).

Media Access Test symbol received on channel B during the preceding symbol window. Updated by the CC for each channel at the end of the symbol window.

When this bit is set to ‘1’, also interrupt flag FLXA0FRSIR.MTSB is set to ‘1’.

Reset when leaving CONFIG state or when entering STARTUP state.

(6) FLXA0FRSWNIT.MTSA

Indicates a MTS Received on Channel A Flag (vSS!ValidMTSA).

Media Access Test symbol received on channel A during the preceding symbol window. Updated by the CC for each channel at the end of the symbol window.

When this bit is set to ‘1’, also interrupt flag FLXA0FRSIR.MTSA is set to ‘1’.

Reset when leaving CONFIG state or when entering STARTUP state.

(7) FLXA0FRSWNIT.TCSB

Indicates a Transmission Conflict in Symbol Window Channel B Flag (vSS!TxConflictB).

Reset when leaving CONFIG state or when entering STARTUP state.

(8) FLXA0FRSWNIT.SBSB

Indicates a Slot Boundary Violation in Symbol Window Channel B Flag (vSS!BViolationB).

Reset when leaving CONFIG state or when entering STARTUP state.

(9) FLXA0FRSWNIT.SESB

Indicates a Syntax Error in Symbol Window Channel B Flag (vSS!SyntaxErrorB).

Reset when leaving CONFIG state or when entering STARTUP state.

(10) FLXA0FRSWNIT.TCSA

Indicates a Transmission Conflict in Symbol Window Channel A Flag (vSS!TxConflictA).

Reset when leaving CONFIG state or when entering STARTUP state.

(11) FLXA0FRSWNIT.SBSA

Indicates a Slot Boundary Violation in Symbol Window Channel A Flag (vSS!BViolationA).

Reset when leaving CONFIG state or when entering STARTUP state.

(12) FLXA0FRSWNIT.SESA

Indicates a Syntax Error in Symbol Window Channel A Flag (vSS!SyntaxErrorA).

Reset when leaving CONFIG state or when entering STARTUP state.

17.2.7.9 FLXA0FRACS — FlexRay Aggregated Channel Status Register

Do not rewrite this register using bit-manipulation instruction.

The aggregated channel status provides the host with an accrued status of channel activity for all communication slots regardless of whether they are assigned for transmission or subscribed for reception.

The aggregated channel status also includes status data from the symbol window and the network idle time.

The status data is updated (set) after each slot and aggregated until it is reset by the host.

During startup the status data is not updated.

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: ERAY + 0128_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	SBVB	CIB	CEDB	SEDB	VFRB	—	—	—	SBVA	CIA	CEDA	SEDA	VFRA
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Table 17.47 FLXA0FRACS Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 13	—	Reserved These bits are always read as 0. When writing, always write 0.
12	SBVB	Slot Boundary Violation on Channel B Flag 0: No slot boundary violation observed 1: Slot boundary violation(s) observed on channel B
11	CIB	Communication Indicator Channel B Flag 0: No valid frame(s) received in slots containing any additional communication 1: Valid frame(s) received on channel B in slots containing any additional communication
10	CEDB	Content Error Detected on Channel B Flag 0: No frame with content error received 1: Frame(s) with content error received on channel B
9	SEDB	Syntax Error Detected on Channel B Flag 0: No syntax error observed 1: Syntax error(s) observed on channel B
8	VFRB	Valid Frame Received on Channel B Flag 0: No valid frame received 1: Valid frame(s) received on channel B
7 to 5	—	Reserved These bits are always read as 0. When writing, always write 0.
4	SBVA	Slot Boundary Violation on Channel A Flag 0: No slot boundary violation observed 1: Slot boundary violation(s) observed on channel A

Table 17.47 FLXA0FRACS Register Contents (2/2)

Bit Position	Bit Name	Function
3	CIA	Communication Indicator Channel A Flag 0: No valid frame(s) received in slots containing any additional communication 1: Valid frame(s) received on channel A in slots containing any additional communication
2	CEDA	Content Error Detected on Channel A Flag 0: No frame with content error received 1: Frame(s) with content error received on channel A
1	SEDA	Syntax Error Detected on Channel A Flag 0: No syntax error observed 1: Syntax error(s) observed on channel A
0	VFRA	Valid Frame Received on Channel A Flag 0: No valid frame received 1: Valid frame(s) received on channel A

(1) FLXA0FRACS.SBVB

Slot Boundary Violation on Channel B Flag

Writing ‘0’ has no effect on the bit value.

This bit is cleared when writing a ‘1’ to it.

One or more slot boundary violations were observed on channel B at any time during the observation period (static or dynamic slots, symbol window, and NIT).

When this flag changes from ‘0’ to ‘1’, interrupt flag FLXA0FREIR.EDB is set to ‘1’.

Reset when leaving CONFIG state or when entering STARTUP state.

(2) FLXA0FRACS.CIB

Communication Indicator Channel B Flag

Writing ‘0’ has no effect on the bit value.

This bit is cleared when writing a ‘1’ to it.

One or more valid frames were received on channel B in slots that also contained any additional communication during the observation period, i.e. one or more slots received a valid frame AND had any combination of either syntax error OR content error OR slot boundary violation.

When this flag changes from ‘0’ to ‘1’, interrupt flag FLXA0FREIR.EDB is set to ‘1’.

Reset when leaving CONFIG state or when entering STARTUP state.

NOTE

The set condition of the flag FLXA0FRACS.CIB is also fulfilled if there is only one single frame in the slot and the slot boundary at the end of the slot is reached during the frames channel idle recognition phase.

(3) FLXA0FRACS.CEDB

Content Error Detected on Channel B Flag

Writing ‘0’ has no effect on the bit value.

This bit is cleared when writing a ‘1’ to it.

One or more frames with a content error were received on channel B in any static or dynamic slot during the observation period.

When this flag changes from ‘0’ to ‘1’, interrupt flag FLXA0FREIR.EDB is set to ‘1’.

Reset when leaving CONFIG state or when entering STARTUP state.

(4) FLXA0FRACS.SEDB

Syntax Error Detected on Channel B Flag

Writing ‘0’ has no effect on the bit value.

This bit is cleared when writing a ‘1’ to it.

One or more syntax errors in static or dynamic slots, symbol window, and NIT were observed on channel B.

When this flag changes from ‘0’ to ‘1’, interrupt flag FLXA0FREIR.EDB is set to ‘1’.

Reset when leaving CONFIG state or when entering STARTUP state.

(5) FLXA0FRACS.VFRB

Valid Frame Received on Channel B Flag

Writing ‘0’ has no effect on the bit value.

This bit is cleared when writing a ‘1’ to it.

One or more valid frames were received on channel B in any static or dynamic slot during the observation period.

Reset when leaving CONFIG state or when entering STARTUP state.

(6) FLXA0FRACS.SBVA

Slot Boundary Violation on Channel A Flag

Writing ‘0’ has no effect on the bit value.

This bit is cleared when writing a ‘1’ to it.

Slot boundary violations were observed on channel A at any time during the observation period (static or dynamic slots, symbol window, and NIT).

When this flag changes from ‘0’ to ‘1’, interrupt flag FLXA0FREIR.EDB is set to ‘1’.

Reset when leaving CONFIG state or when entering STARTUP state.

(7) FLXA0FRACS.CIA

Communication Indicator Channel A Flag

Writing '0' has no effect on the bit value.

This bit is cleared when writing a ‘1’ to it.

One or more valid frames were received on channel A in slots that also contained any additional communication during the observation period, i.e. one or more slots received a valid frame, and had one of syntax error, content error, or slot boundary violation.

When this flag changes from ‘0’ to ‘1’, interrupt flag FLXA0FREIR.EDB is set to ‘1’.

Reset when leaving CONFIG state or when entering STARTUP state.

NOTE

The set condition of the flag FLXA0FRACS.CIA is also fulfilled if there is only one single frame in the slot and the slot boundary at the end of the slot is reached during the frames channel idle recognition phase.

(8) FLXA0FRACS.CEDA

Content Error Detected on Channel A Flag

Writing '0' has no effect on the bit value.

This bit is cleared when writing a ‘1’ to it.

One or more frames with a content error were received on channel A in any static or dynamic slot during the observation period.

When this flag changes from ‘0’ to ‘1’, interrupt flag FLXA0FREIR.EDB is set to ‘1’.

Reset when leaving CONFIG state or when entering STARTUP state.

(9) FLXA0FRACS.SEDA

Syntax Error Detected on Channel A Flag

Writing '0' has no effect on the bit value.

This bit is cleared when writing a ‘1’ to it.

One or more syntax errors in static or dynamic slots, symbol window, and NIT were observed on channel A.

When this flag changes from ‘0’ to ‘1’, interrupt flag FLXA0FREIR.EDB is set to ‘1’.

Reset when leaving CONFIG state or when entering STARTUP state.

(10) FLXA0FRACS.VFRA

Valid Frame Received on Channel A Flag

Writing '0' has no effect on the bit value.

This bit is cleared when writing a ‘1’ to it.

One or more valid frames were received on channel A in any static or dynamic slot during the observation period.

Reset when leaving CONFIG state or when entering STARTUP state.

17.2.7.10 FLXA0FRESIDn — FlexRay Even Sync ID Register n (n = 1 to 15)

Registers FLXA0FRESID1 to FLXA0FRESID15 hold the frame IDs of the sync frames received in even communication cycles used for clock synchronization up to the limit of gSyncNodeMax. The values are sorted in ascending order, with register FLXA0FRESID1 holding the lowest received sync frame ID. If the node itself transmits a sync frame in an even communication cycle, register FLXA0FRESID1 holds the respective sync frame ID as configured in message buffer 0 and flags FLXA0FRESID1.RXEA, FLXA0FRESID1.RXEB are set. The value is updated during the NIT of each even communication cycle.

Access: This register can be read in 8-, 16-, or 32-bit units.

Address: ERAY + 0130_H to ERAY + 0168_H (ERAY + 0130_H + (n-1)*4)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RXEB	RXEA	—	—	—	—	EID[9:0]									
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 17.48 FLXA0FRESIDn (n = 1 to 15) Register Contents

Bit Position	Bit Name	Function
31 to 16	—	Reserved These bits are always read as 0. When writing, always write 0.
15	RXEB	Received/Configured Even Sync ID on Channel B Flag 0: No sync frame received on channel B or node not configured to transmit sync frames 1: Sync frame received on channel B or node configured to transmit sync frames
14	RXEA	Received/Configured Even Sync ID on Channel A Flag 0: No sync frame received on channel A or node not configured to transmit sync frames 1: Sync frame received on channel A or node configured to transmit sync frames
13 to 10	—	Reserved These bits are always read as 0. When writing, always write 0.
9 to 0	EID[9:0]	Even Sync ID Flags (vsSyncIDListA,B even)

(1) FLXA0FRESIDn.RXEB

Received/Configured Even Sync ID on Channel B Flag

Signals that a sync frame corresponding to the stored even sync ID was received on channel B or that the node is configured to be a sync node with key slot indicated by the FLXA0FRESID1.EID.

Reset when leaving CONFIG state or when entering STARTUP state.

(2) FLXA0FRESIDn.RXEA

Received/Configured Even Sync ID on Channel A Flag

Signals that a sync frame corresponding to the stored even sync ID was received on channel A or that the node is configured to be a sync node with key slot indicated by the FLXA0FRESID1.EID.

Reset when leaving CONFIG state or when entering STARTUP state.

(3) FLXA0FRESIDn.EID

Even Sync ID Flags (vsSyncIDListA, B even)

Sync frame ID even communication cycle

Reset when leaving CONFIG state or when entering STARTUP state.

17.2.7.11 FLXA0FROSIDn — FlexRay Odd Sync ID Register n (n = 1 to 15)

Registers FLXA0FROSID1 to FLXA0FROSID15 hold the frame IDs of the sync frames received in odd communication cycles used for clock synchronization up to the limit of gSyncNodeMax. The values are sorted in ascending order, with register FLXA0FROSID1 holding the lowest received sync frame ID. If the node itself transmits a sync frame in an odd communication cycle, register FLXA0FROSID1 holds the respective sync frame ID as configured in message buffer 0 and flags FLXA0FROSID1.RXOA and FLXA0FROSID1.RXOB are set. The value is updated during the NIT of each odd communication cycle.

Access: This register can be read in 8-, 16-, or 32-bit units.

Address: ERAY + 0170_H to ERAY + $01A8_H$ (ERAY + 0170_H + (n-1)*4)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RXOB	RXOA	—	—	—	—	OID[9:0]									
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 17.49 FLXA0FROSIDn Register Contents

Bit Position	Bit Name	Function
31 to 16	—	Reserved These bits are always read as 0. When writing, always write 0.
15	RXOB	Received/Configured Odd Sync ID on Channel B Flag 0: No sync frame received on channel B or node not configured to transmit sync frames 1: Sync frame received on channel B or node configured to transmit sync frames
14	RXOA	Received/Configured Odd Sync ID on Channel A Flag 0: No sync frame received on channel A or node not configured to transmit sync frames 1: Sync frame received on channel A or node configured to transmit sync frames
13 to 10	—	Reserved These bits are always read as 0. When writing, always write 0.
9 to 0	OID[9:0]	Odd Sync ID Flags (vsSyncIDListA,B odd)

(1) FLXA0FROSIDn.RXOB

Received/Configured Odd Sync ID on Channel B Flag

Signals that a sync frame corresponding to the stored odd sync ID was received on channel B or that the node is configured to be a sync node with key slot = FLXA0FROSID1.OID.

Reset when leaving CONFIG state or when entering STARTUP state.

(2) FLXA0FROSIDn.RXOA

Received/Configured Odd Sync ID on Channel A Flag

Signals that a sync frame corresponding to the stored odd sync ID was received on channel A or that the node is configured to be a sync node with key slot = FLXA0FRISID1.OID.

Reset when leaving CONFIG state or when entering STARTUP state.

(3) FLXA0FROSIDn.OID

Odd Sync ID Flags (vsSyncIDListA,B odd)

Sync frame ID odd communication cycle.

Reset when leaving CONFIG state or when entering STARTUP state.

17.2.7.12 FLXA0FRNMVn — FlexRay Network Management Vector Register n (n = 1 to 3)

The three network management registers hold the accrued NM vector (see **Section 17.3.7, Network Management**).

The CC updates the NM vector at the end of each communication cycle as long as the CC is either in NORMAL_ACTIVE or NORMAL_PASSIVE state.

NMVn-bytes exceeding the configured NM vector length are not valid.

For information about the byte alignment of the received NM vector in this register see **Section 17.3.17, Byte Alignment**.

Access: This register can be read in 8-, 16-, or 32-bit units.

Address: ERAY + $01B0_H$ to ERAY + $01B8_H$ (ERAY + $01B0_H$ + (n-1)*4)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	NM[31:16]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	NM[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 17.50 FLXA0FRNMVn Register Contents

Bit Position	Bit Name	Function
31 to 0	NM[31:0]	NM Vector the three network management vector registers hold the accrued NM vector (configurable 0 to 12 bytes). The NM vector to be held is generated by bit-wise logic OR for each NM vector received on each channel (valid static frames with PPI = '1') (see **Section 17.3.7, Network Management**). For information about the byte alignment of the received NM vector in this register, see **Section 17.3.17, Byte Alignment**. NMVn-bytes exceeding the configured NM vector length are not valid. The register contents are updated at the end of each communication cycle as long as the CC is either in NORMAL_ACTIVE or NORMAL_PASSIVE state. These bits are cleared when leaving CONFIG state or when entering STARTUP state.

17.2.8 Message Buffer Control Registers

17.2.8.1 FLXA0FRMRC — FlexRay Message RAM Configuration Register

The message RAM configuration register defines the number of message buffers assigned to the static segment, dynamic segment, and FIFO.

The message RAM can be divided into up three different areas; static buffer area, static and dynamic buffer area, FIFO area. If present, the static buffer area is starting at message buffer 0.

The start of the static and dynamic buffer area is configured by FLXA0FRMRC.FDB. FLXA0FRMRC.FDB defines the end of the static buffer area. If no static buffer area is present, the static and dynamic buffer area starts at message buffer 0.

The start of the FIFO area is configured by FLXA0FRMRC.FFB. FLXA0FRMRC.FFB defines the end of the previous area, which can be either the static buffer area or the static and dynamic buffer area.

If no static buffer area and no static and dynamic buffer area is present, the FIFO area starts at message buffer 0.

With FLXA0FRMRC.LCB, the end of the last configured area is configured which can be the static buffer area, the static and dynamic buffer area, or the FIFO area.

Figure 17.2 shows an example configuration of the message RAM where all there areas are configured.

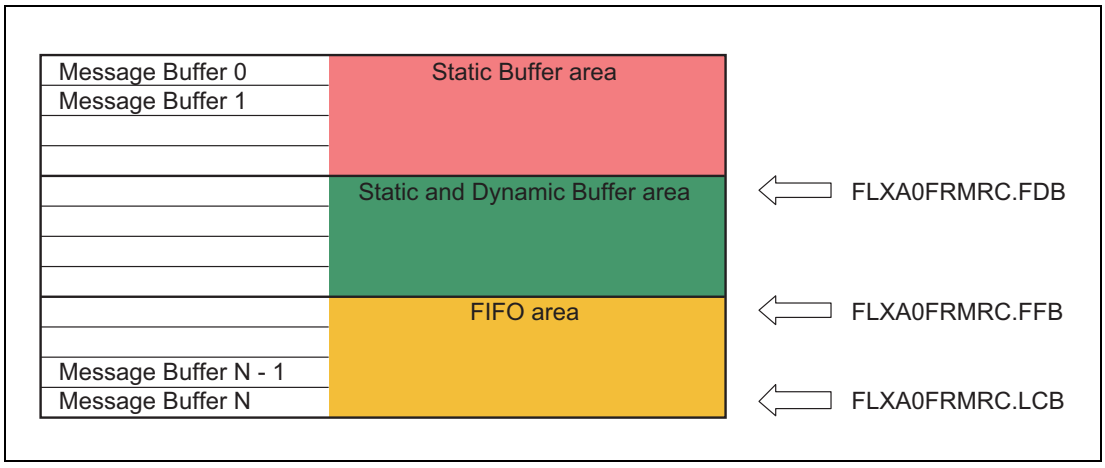


Figure 17.2 Message RAM Organization

NOTES

1. In case the node is configured as sync node (FLXA0FRSUCC1.TXSY = ‘1’) or for single slot mode operation (FLXA0FRSUCC1.TSM = ‘1’), message buffer 0 and 1 are reserved for sync frames or single slot frames and have to be configured with the node-specific key slot ID. In case the node is neither configured as sync node nor for single slot operation message buffer 0 and 1 are treated like all other message buffers.
2. The maximum number of header sections is 128. This means a maximum of 128 message buffers can be configured. The maximum length of a data section is 254 bytes. The length of the data section can be configured separately for each message buffer. For details see **Section 17.3.13, Message RAM**.
3. In case two or more message buffers are assigned to slot 1 by use of cycle filtering, all of them must be located either in the “static buffers” or at the beginning of the “static + dynamic buffers” section.

4. The FlexRay protocol specification requires that each node has to send a frame in its key slot. Therefore at least message buffer 0 is reserved for transmission in the key slot. Due to this requirement a maximum number of 127 message buffers can be assigned to the FIFO. Nevertheless, a non-protocol-conforming configuration without a transmission slot in the static segment would still be operational.
5. The payload length and the length of the data section need to be configured identical for all message buffers belonging to the FIFO via FLXA0FRWRHS2.PLC and FLXA0FRWRHS3.DP. When the CC is not in DEFAULT_CONFIG or CONFIG state reconfiguration of message buffers belonging to the FIFO is locked.

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: ERAY + 0300_H

Value after reset: $0180\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	SPLM	SEC[1:0]		LCB[7:0]							
Value after reset	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	FFB[7:0]								FDB[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 17.51 FLXA0FRMRC Register Contents

Bit Position	Bit Name	Function
31 to 27	—	Reserved These bits are always read as 0. When writing, always write 0.
26	SPLM	Sync Frame Payload Multiplex 0: Writing to message buffer 0 is prohibited. 1: Writing to message buffer 0 or 1 is prohibited.
25, 24	SEC[1:0]	Secure Buffers 00: Writing is enabled for all buffers. 01: Writing is prohibited to static buffer and FIFO, limited transmission 10: Writing is prohibited for all buffers. 11: Writing is prohibited for all buffers, limited transmission
23 to 16	LCB[7:0]	Last Configured Buffer 0 to 127: Number of message buffers is FLXA0FRMRF.LCB + 1 128: No message buffer configured
15 to 8	FFB[7:0]	First Buffer of FIFO 0: All message buffers assigned to the FIFO 1 to 127: Message buffers from FLXA0FRMRC.FFB to FLXA0FRMRC.LCB assigned to the FIFO 128: No message buffer configured
7 to 0	FDB[7:0]	First Dynamic Buffer 0: No group of message buffers exclusively for the static segment configured 1 to 127: Message buffers 0 to FLXA0FRMRC.FDB - 1 reserved for static segment 128: No dynamic message buffers configured

(1) FLXA0FRMRC.SPLM

Sync Frame Payload Multiplex

The user can only write to this bit when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

This bit is only enabled if the node is configured as sync node (FLXA0FRSUCC1.TXSY = ‘1’) or for single slot mode operation (FLXA0FRSUCC1.TSM = ‘1’).

When this bit is set to ‘1’, message buffers 0 and 1 are dedicated for sync frame transmission with separate payload data on channel A and B.

When this bit is set to ‘0’, sync frames are transmitted from message buffer 0 with the same payload data on all channels configured. Note that the channel filter configuration for message buffer 0 and message buffer 1 are selected according to this SPLM bit setting.

(2) FLXA0FRMRC.SEC

Secure Buffer

The user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

These bits are disabled when the CC is in DEFAULT_CONFIG or CONFIG state.

For temporary unlocking, see **Section 17.3.13.4, Host Handling of Access Errors**.

00 = Writing is enabled for any buffers.

Reconfiguration of message buffers enabled with numbers < FLXA0FRMRC.FFB enabled

Exception: In nodes configured for sync frame transmission or for single slot mode operation, writing to message buffer 0 (and if FLXA0FRMRC.SPLM = '1', also message buffer 1) is prohibited.

01 = static buffers locked, FIFO locked, limited transmission

Reconfiguration of message buffers with numbers < FLXA0FRMRC.FDB and with numbers ≥ FLXA0FRMRC.FFB locked and transmission of message buffers for static segment with numbers ≥ FLXA0FRMRC.FDB disabled

10 = all buffers locked

Writing to any message buffers is prohibited.

11 = Writing to any buffers is prohibited.

Reconfiguration of all message buffers locked and transmission of message buffers for static segment with numbers ≥ FLXA0FRMRC.FDB disabled

(3) FLXA0FRMRC.LCB

Last Buffer Setting

The user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

When a Static and Dynamic Buffer area is configured (FLXA0FRMRC.FDB < 128), the user should configure FLXA0FRMRC.LCB ≥ FLXA0FRMRC.FDB.

When a FIFO area is configured (FLXA0FRMRC.FFB < 128), the user should configure FLXA0FRMRC.LCB ≥ FLXA0FRMRC.FFB.

(4) FLXA0FRMRC.FFB

FIFO First Buffer Setting

The user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

When a Static and Dynamic Buffer area is configured (FLXA0FRMRC.FDB < 128), the user should configure FLXA0FRMRC.FFB > FLXA0FRMRC.FDB.

(5) FLXA0FRMRC.FDB

First Dynamic Buffer Setting

The user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

17.2.8.2 FLXA0FRFRF — FlexRay FIFO Rejection Filter Register

The FIFO Rejection Filter defines a user specified sequence of bits to which channel, frame ID, and cycle count of the incoming frames are compared. Together with the FIFO Rejection Filter Mask Register this register determines whether a message is rejected by the FIFO.

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: ERAY + 0304_H

Value after reset: $0180\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	RNF	RSS	CYF[6:0]						
Value after reset	0	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	FID[10:0]											CH[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 17.52 FLXA0FRFRF Register Contents

Bit Position	Bit Name	Function
31 to 25	—	Reserved These bits are always read as 0. When writing, always write 0.
24	RNF	Reject Null Frames 0: Null frames are stored in the FIFO 1: Reject all null frames
23	RSS	Reject in Static Segment 0: FIFO also used for static segment 1: Reject messages in static segment
22 to 16	CYF[6:0]	Cycle Counter Filter
15 to 13	—	Reserved These bits are always read as 0. When writing, always write 0.
12 to 2	FID[10:0]	Frame ID Filter 0 to 2047: Frame ID filter values
1, 0	CH[1:0]	Channel Filter 00: receive on both channels 01: receive only on channel B 10: receive only on channel A 11: Receive prohibited

(1) FLXA0FRFRF.RNF

Reject Null Frame

The user can only write to this bit when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

If this bit is set to ‘1’, received null frames are not stored in the FIFO.

(2) FLXA0FRFRF.RSS

Reject in Static Segment

The user can only write to this bit when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

If this bit is set to '1', the FIFO is used only for the dynamic segment.

(3) FLXA0FRFRF.CYF

Cycle Counter Filter

The user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

The 7-bit cycle counter filter determines the cycle set to which frame ID and channel rejection filter are applied. In cycles that are not specified by FLXA0FRFRF.CYF, all frames are rejected. For details about the configuration of the cycle counter filter see **Section 17.3.8.2, Cycle Counter Filtering**.

(4) FLXA0FRFRF.FID

Frame ID Filter

The user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

Determines the frame ID to be rejected by the FIFO. With the additional configuration of register FLXA0FRFRFM, the corresponding frame ID filter bits are ignored. When FLXA0FRFRFM.MFID is zero, a frame ID filter value of zero means that there is no frame ID to be rejected.

(5) FLXA0FRFRF.CH

Channel Filter

The user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

If reception on both channels is configured, also in static segment always both frames (from channel A and B) are stored in the FIFO, even if they are identical.

17.2.8.3 FLXA0FRFRFM — FlexRay FIFO Rejection Filter Mask Register

The FlexRay FIFO Rejection Filter Mask Register specifies frame ID filter bits relevant to rejection filtering. If a bit is set to '1', it indicates that the corresponding bit in the FLXA0FRFRF register will not be considered for rejection filtering.

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: ERAY + 0308_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	MFID[10:0]											—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Table 17.53 FLXA0FRFRFM Register Contents

Bit Position	Bit Name	Function
31 to 13	—	Reserved These bits are always read as 0. When writing, always write 0.
12 to 2	MFID[10:0]	Mask Frame ID Filter 0: Corresponding frame ID filter bit is used for rejection filtering 1: Ignore corresponding frame ID filter bit.
1, 0	—	Reserved These bits are always read as 0. When writing, always write 0.

(1) FLXA0FRFRFM.MFID

Mask Frame ID Filter

The user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

17.2.8.4 FLXA0FRFCL — FlexRay FIFO Critical Level Register

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: ERAY + $030C_H$

Value after reset: $0000\ 0080_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	CL[7:0]							
Value after reset	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 17.54 FLXA0FRFCL Register Contents

Bit Position	Bit Name	Function
31 to 8	—	Reserved These bits are always read as 0. When writing, always write 0.
7 to 0	CL[7:0]	Critical Level Setting Critical level setting

(1) FLXA0FRFCL.CL

Critical Level Setting

The user can only write to these bits when FLXA0FRCCSV.POCS is DEFAULT_CONFIG or CONFIG.

When the receive FIFO fill level FLXA0FRFSR.RFFL is equal or greater than the critical level configured by FLXA0FRFCL.CL, the receive FIFO critical level flag FLXA0FRFSR.RFCL is set to 1.

If FLXA0FRFCL.CL is programmed to values > 128, bit FLXA0FRFSR.RFCL is never set to 1.

17.2.9 Message Buffer Status Registers

17.2.9.1 FLXA0FRMHDS — FlexRay Message Handler Status Register

Do not rewrite this register using bit-manipulation instruction.

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: ERAY + 0310_H

Value after reset: $0000\ 0080_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	MBU[6:0]							—	MBT[6:0]						
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	FMB[6:0]							CRAM	MFMB	FMBD	ATBF2	ATBF1	AMR	—	—
Value after reset	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R

Table 17.55 FLXA0FRMHDS Register Contents

Bit Position	Bit Name	Function
31	—	Reserved This bit is always read as 0. When writing, always write 0.
30 to 24	MBU[6:0]	Message Buffer Updated Flags
23	—	Reserved This bit is always read as 0. When writing, always write 0.
22 to 16	MBT[6:0]	Message Buffer Transmitted Flags
15	—	Reserved This bit is always read as 0. When writing, always write 0.
14 to 8	FMB[6:0]	Faulty Message Buffer Number Flags
7	CRAM	Internal RAM Clear Flag 0: No execution of the CHI command CLEAR_RAMS 1: Execution of the CHI command CLEAR_RAMS ongoing
6	MFMB	Multiple Faulty Message Buffer Detection Flag 0: No additional faulty message buffer 1: Additional faulty message buffer was detected while the FMBD flag is set to 1.
5	FMBD	Faulty Message Buffer Detection Flag 0: No faulty message buffer 1: Message buffer referenced by FLXA0FRMHDS.FMB holds faulty data with a parity error.
4	ATBF2	Temporary Buffer RAM B Access Error Flag 0: No access error 1: Access error occurred when reading the RAM B.
3	ATBF1	Temporary Buffer RAM A Access Error Flag 0: No access error 1: Access error occurred when reading the RAM A.
2	AMR	Message RAM Access Error Flag 0: No access error 1: Access error occurred when reading the message RAM.
1, 0	—	Reserved These bits are always read as 0. When writing, always write 0

(1) FLXA0FRMHDS.MBU

Message Buffer Updated Flags

These flags indicate the number of message buffer that was updated last by the CC. For this message buffer the respective ND and/or MBC flag in the FLXA0FRNDAT1/2/3/4 registers and the FLXA0FRMBSC1/2/3/4 registers are also set to ‘1’.

Reset when leaving CONFIG state or when entering STARTUP state.

Reset by the CHI command CLEAR_RAMS.

(2) FLXA0FRMHDS.MBT

Message Buffer Transmitted Flags

These flags indicate the number of last successfully transmitted message buffer.

If the message buffer is configured for single-shot mode, the respective TXR flag in the FLXA0FRTXRQ1/2/3/4 registers are reset to ‘0’.

Reset when leaving CONFIG state or when entering STARTUP state.

Reset by the CHI command CLEAR_RAMS.

(3) FLXA0FRMHDS.FMB

Faulty Message Buffer Number Flags

These flags indicate that an access error occurred when reading from the message buffer referenced by FLXA0FRMHDS.FMB.

The value of this flag is only valid whenever one of AMR flag, ATBF1 flag, ATBF2 flag, or FMBD flag in the FLXA0FRMHDS register is set to 1.

This flag is not updated while the FLXA0FRMHDS.FMBD flag is 1.

This flag is cleared by the CHI command CLEAR_RAMS.

(4) FLXA0FRMHDS.CRAM

Internal RAM Clear Flag

This flag indicates that the CHI command CLEAR_RAMS is ongoing (all bits of the message RAM, input buffer, output buffer and temporary buffer RAM are written to ‘0’).

This flag is set by the CHI command CLEAR_RAMS.

(5) FLXA0FRMHDS.MFMB

Multiple Faulty Message Buffer Detection Flag

Writing '0' has no effect on the bit value.

This bit is cleared when writing ‘1’ to it.

This bit indicates that an additional faulty message buffer was detected while the FMBD flag is set.

This bit is cleared by the CHI command CLEAR_RAMS.

(6) FLXA0FRMHDS.FMBD

Faulty Message Buffer Detection Flag

Writing '0' has no effect on the bit value.

This bit is cleared when writing ‘1’ to it.

This bit indicates that the message buffer referenced by FLXA0FRMHDS.FMB holds faulty data due to an access error.

This bit is cleared by the CHI command CLEAR_RAMS.

(7) FLXA0FRMHDS.ATBF2

Temporary Buffer RAM B Access Error Flag

Writing ‘0’ has no effect on the bit value.

This bit is cleared when writing ‘1’ to it.

This flag indicates that an access error occurred when reading the temporary buffer RAM B.

NOTE

When this flag changes from ‘0’ to ‘1’, the AERR bit in the FLXA0FREIR register is set to ‘1’. This flag can be reset by the CHI command CLEAR_RAMS.

(8) FLXA0FRMHDS.ATBF1

Temporary Buffer RAM A Access Error Flag

Writing ‘0’ has no effect on the bit value.

This bit is cleared when writing ‘1’ to it.

This flag indicates that an access error occurred when reading the temporary buffer RAM A.

NOTE

When this flag changes from ‘0’ to ‘1’, the AERR bit in the FLXA0FREIR register is set to ‘1’. This flag can be reset by the CHI command CLEAR_RAMS.

(9) FLXA0FRMHDS.AMR

Message RAM Access Error Flag

Writing ‘0’ has no effect on the bit value.

This bit is cleared when writing ‘1’ to it.

This flag indicates that an access error occurred when reading the message RAM.

NOTE

When this flag changes from ‘0’ to ‘1’, the AERR bit in the FLXA0FREIR register is set to ‘1’. This flag can be reset by the CHI command CLEAR_RAMS.

17.2.9.2 FLXA0FRLDTS — FlexRay Last Dynamic Transmit Slot Register

Access: This register can be read in 8-, 16-, or 32-bit units.

Address: ERAY + 0314_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	LDTB[10:0]										
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	LDTA[10:0]										
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 17.56 FLXA0FRLDTS Register Contents

Bit Position	Bit Name	Function
31 to 27	—	Reserved These bits are always read as 0. When writing, always write 0.
26 to 16	LDTB[10:0]	Last Dynamic Transmission Channel B Flags
15 to 11	—	Reserved These bits are always read as 0. When writing, always write 0.
10 to 0	LDTA[10:0]	Last Dynamic Transmission Channel A Flags

(1) FLXA0FRLDTS.LDTB

Last Dynamic Transmission Channel B Flags

Store the value of vSlotCounter at the time of the last frame transmission on channel B in the dynamic segment.

It is updated at the end of the dynamic segment and is reset to zero if no frame was transmitted during the dynamic segment.

Reset when leaving CONFIG state or when entering STARTUP state.

Reset by the CHI command CLEAR_RAMS.

(2) FLXA0FRLDTS.LDTA

Last Dynamic Transmission Channel A Flags

Store the value of vSlotCounter at the time of the last frame transmission on channel A in the dynamic segment.

It is updated at the end of the dynamic segment and is reset to zero if no frame was transmitted during the dynamic segment.

Reset when leaving CONFIG state or when entering STARTUP state.

Reset by the CHI command CLEAR_RAMS.

17.2.9.3 FLXA0FRFSR — FlexRay FIFO Status Register

Access: This register can be read in 8-, 16-, or 32-bit units.

Address: ERAY + 0318_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RFFL[7:0]								—	—	—	—	—	RFO	RFCL	RFNE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 17.57 FLXA0FRFSR Register Contents

Bit Position	Bit Name	Function
31 to 16	—	Reserved These bits are always read as 0. When writing, always write 0.
15 to 8	RFFL[7:0]	Receive FIFO Fill Level Flags
7 to 3	—	Reserved These bits are always read as 0. When writing, always write 0.
2	RFO	Receive FIFO Overrun Flag 0: No receive FIFO overrun is detected 1: A receive FIFO overrun has been detected
1	RFCL	Receive FIFO Critical Level Flag 0: Receive FIFO is below critical level 1: Receive FIFO critical level is reached
0	RFNE	Receive FIFO Not Empty Flag 0: Receive FIFO is empty 1: Receive FIFO is not empty

(1) FLXA0FRFSR.RFFL

Receive FIFO Fill Level Flags

Indicate the number of FIFO buffers filled with new data not yet read by the host. Maximum value is 128.

Reset when leaving CONFIG state or when entering STARTUP state.

Reset by the CHI command CLEAR_RAMS.

(2) FLXA0FRFSR.RFO

Receive FIFO Overrun Flag

The flag is set to ‘1’ by the CC when a receive FIFO overrun is detected.

When a receive FIFO overrun occurs, the oldest message is overwritten with the actual received message. In addition, interrupt flag FLXA0FREIR.RFO is set to ‘1’.

The flag is cleared when the FIFO is read.

Reset when leaving CONFIG state or when entering STARTUP state.

Reset by the CHI command CLEAR_RAMS.

(3) FLXA0FRFSR.RFCL

Receive FIFO Critical Level Flag

This flag is set to ‘1’ when the receive FIFO fill level FLXA0FRFSR.RFFL is equal or greater than the critical level as configured by FLXA0FRFCL.CL.

When FLXA0FRFSR.RFCL changes from ‘0’ to ‘1’ bit FLXA0FRSIR.RFCL is set to ‘1’, and if enabled, an interrupt is generated.

The flag is cleared by the CC as soon as FLXA0FRFSR.RFFL drops below FLXA0FRFCL.CL.

Reset when leaving CONFIG state or when entering STARTUP state.

Reset by the CHI command CLEAR_RAMS.

(4) FLXA0FRFSR.RFNE

Receive FIFO Not Empty Flag

This flag is set to ‘1’ by the CC when a received valid frame (data or null frame depending on rejection mask) was stored in the FIFO. In addition, interrupt flag FLXA0FRSIR.RFNE is set to ‘1’.

The bit is reset to ‘0’ after the host has read all messages from the FIFO.

Reset when leaving CONFIG state or when entering STARTUP state.

Reset by the CHI command CLEAR_RAMS.

17.2.9.4 FLXA0FRMHDF — FlexRay Message Handler Constraints Flags Register

Do not rewrite this register using bit-manipulation instruction.

Some constraints exist for the message handler regarding bus clock frequency, message RAM configuration, and FlexRay bus traffic. To simplify software development, constraints violations are reported by setting flags in the FLXA0FRMHDF.

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: ERAY + $031C_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	WAHP	TNSB	TNSA	TBFB	TBFA	FNFB	FNFA	SNUB	SNUA
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 17.58 FLXA0FRMHDF Register Contents

Bit Position	Bit Name	Function
31 to 9	—	Reserved These bits are always read as 0. When writing, always write 0.
8	WAHP	Write Attempt to Header Partition Flag 0: No write attempt to header partition 1: A write attempt to header partition
7	TNSB	Transmission Not Started Channel B Flag 0: No transmission not started on channel B 1: Transmission not started on channel B
6	TNSA	Transmission Not Started Channel A Flag 0: No transmission not started on channel A 1: Transmission not started on channel A
5	TBFB	Temporary Buffer Access Failure B Flag 0: No TBF B access failure 1: TBF B access failure
4	TBFA	Temporary Buffer Access Failure A Flag 0: No TBF A access failure 1: TBF A access failure
3	FNFB	Find Sequence Not Finished Channel B Flag 0: No find sequence not finished for channel B 1: Find sequence not finished for channel B
2	FNFA	Find Sequence Not Finished Channel A Flag 0: No find sequence not finished for channel A 1: Find sequence not finished for channel A
1	SNUB	Status Not Updated Channel B Flag 0: No overload condition occurred when updating MBS for channel B 1: Message buffer status (FLXA0FRMBS) for channel B not updated
0	SNUA	Status Not Updated Channel A Flag 0: No overload condition occurred when updating MBS for channel A 1: Message buffer status (FLXA0FRMBS) for channel A not updated

(1) FLXA0FRMHDF.WAHP

Write Attempt to Header Partition Flag

Writing ‘0’ has no effect on the bit value.

This bit is cleared when writing a ‘1’ to it.

Outside DEFAULT_CONFIG and CONFIG state this flag is set to ‘1’ when the message handler tries to write message data into the header partition of the message RAM due to faulty configuration of a message buffer. The write attempt is not executed, to protect the header partition from unintended write accesses.

When this flag changes from ‘0’ to ‘1’ in addition interrupt flag FLXA0FREIR.MHF is set to ‘1’.

Reset when leaving CONFIG state or when entering STARTUP state.

Reset by the CHI command CLEAR_RAMS.

(2) FLXA0FRMHDF.TNSB Flag

Transmission Not Started Channel B

Writing ‘0’ has no effect on the bit value.

This bit is cleared when writing a ‘1’ to it.

This flag is set to ‘1’ when the message handler was not ready to start a scheduled transmission on channel B at the action point of the configured slot.

When this flag changes from ‘0’ to ‘1’, interrupt flag FLXA0FREIR.MHF is set to ‘1’.

Reset when leaving CONFIG state or when entering STARTUP state.

Reset by the CHI command CLEAR_RAMS.

(3) FLXA0FRMHDF.TNSA

Transmission Not Started Channel A Flag

Writing ‘0’ has no effect on the bit value.

This bit is cleared when writing a ‘1’ to it.

This flag is set when the message handler was not ready to start a scheduled transmission on channel A at the action point of the configured slot.

When this flag changes from ‘0’ to ‘1’, interrupt flag FLXA0FREIR.MHF is set to ‘1’.

Reset when leaving CONFIG state or when entering STARTUP state.

Reset by the CHI command CLEAR_RAMS.

(4) FLXA0FRMHDF.TBFB

Temporary Buffer Access Failure B Flag

Writing ‘0’ has no effect on the bit value.

This bit is cleared when writing a ‘1’ to it.

This flag is set to ‘1’ when a read or write access to TBF B requested by PRT (protocol controller) B could not complete within the specified time.

When this flag changes from ‘0’ to ‘1’, interrupt flag FLXA0FREIR.MHF is set to ‘1’.

Reset when leaving CONFIG state or when entering STARTUP state.

Reset by the CHI command CLEAR_RAMS.

(5) FLXA0FRMHDF.TBFA

Temporary Buffer Access Failure A Flag

Writing ‘0’ has no effect on the bit value.

This bit is cleared when writing a ‘1’ to it.

This flag is set to ‘1’ by the CC when a read or write access to TBF A requested by PRT A could not complete within the specified time.

When this flag changes from ‘0’ to ‘1’, interrupt flag FLXA0FREIR.MHF is set to ‘1’.

Reset when leaving CONFIG state or when entering STARTUP state.

Reset by the CHI command CLEAR_RAMS.

(6) FLXA0FRMHDF.FNFB

Find Sequence Not Finished Channel B Flag

Writing ‘0’ has no effect on the bit value.

This bit is cleared when writing a ‘1’ to it.

This flag is set to ‘1’ when the message handler, due to overload condition, was not able to finish a find sequence (scan of message RAM for matching message buffer).

When this flag changes from ‘0’ to ‘1’, interrupt flag FLXA0FREIR.MHF is set to ‘1’.

Reset when leaving CONFIG state or when entering STARTUP state.

Reset by the CHI command CLEAR_RAMS.

(7) FLXA0FRMHDF.FNFA

Find Sequence Not Finished Channel A Flag

Writing ‘0’ has no effect on the bit value.

This bit is cleared when writing a ‘1’ to it.

This flag is set to ‘1’ when the message handler, due to overload condition, was not able to finish a find sequence (scan of message RAM for matching message buffer).

When this flag changes from ‘0’ to ‘1’, interrupt flag FLXA0FREIR.MHF is set to ‘1’.

Reset when leaving CONFIG state or when entering STARTUP state.

Reset by the CHI command CLEAR_RAMS.

(8) FLXA0FRMHDF.SNUB

Status Not Updated Channel B Flag

Writing ‘0’ has no effect on the bit value.

This bit is cleared when writing a ‘1’ to it.

This flag is set to ‘1’ when the message handler, due to overload condition, was not able to update a message buffer's status MBS.

When this flag changes from ‘0’ to ‘1’, interrupt flag FLXA0FREIR.MHF is set to ‘1’.

Reset when leaving CONFIG state or when entering STARTUP state.

Reset by the CHI command CLEAR_RAMS.

(9) FLXA0FRMHDF.SNUA

Status Not Updated Channel A Flag

Writing ‘0’ has no effect on the bit value.

This bit is cleared when writing a ‘1’ to it.

This flag is set to ‘1’ when the message handler, due to overload condition, was not able to update a message buffer's status MBS.

When this flag changes from ‘0’ to ‘1’, interrupt flag FLXA0FREIR.MHF is set to ‘1’.

Reset when leaving CONFIG state or when entering STARTUP state.

Reset by the CHI command CLEAR_RAMS.

17.2.9.5 FLXA0FRTXRQi — FlexRay Transmission Request i (i = 1 to 4)

The four registers reflect the state of the TXR flags of all configured message buffers. The flags are evaluated for transmit buffers only. If the number of configured message buffers is less than 128, the remaining TXR flags have no meaning.

Access: This register can be read in 8-, 16-, or 32-bit units.

Address: ERAY + 0320_H to ERAY to $032C_H$ (ERAY + 0320_H + (i-1)*4)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TXRo (o = (i - 1) × 32 + 31)	TXRo (o = (i - 1) × 32 + 30)	TXRo (o = (i - 1) × 32 + 29)	TXRo (o = (i - 1) × 32 + 28)	TXRo (o = (i - 1) × 32 + 27)	TXRo (o = (i - 1) × 32 + 26)	TXRo (o = (i - 1) × 32 + 25)	TXRo (o = (i - 1) × 32 + 24)	TXRo (o = (i - 1) × 32 + 23)	TXRo (o = (i - 1) × 32 + 22)	TXRo (o = (i - 1) × 32 + 21)	TXRo (o = (i - 1) × 32 + 20)	TXRo (o = (i - 1) × 32 + 19)	TXRo (o = (i - 1) × 32 + 18)	TXRo (o = (i - 1) × 32 + 17)	TXRo (o = (i - 1) × 32 + 16)
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TXRo (o = (i - 1) × 32 + 15)	TXRo (o = (i - 1) × 32 + 14)	TXRo (o = (i - 1) × 32 + 13)	TXRo (o = (i - 1) × 32 + 12)	TXRo (o = (i - 1) × 32 + 11)	TXRo (o = (i - 1) × 32 +10)	TXRo (o = (i - 1) × 32 + 9)	TXRo (o = (i - 1) × 32 + 8)	TXRo (o = (i - 1) × 32 + 7)	TXRo (o = (i - 1) × 32 + 6)	TXRo (o = (i - 1) × 32 + 5)	TXRo (o = (i - 1) × 32 + 4)	TXRo (o = (i - 1) × 32 + 3)	TXRo (o = (i - 1) × 32 + 2)	TXRo (o = (i - 1) × 32 + 1)	TXRo (o = (i - 1) × 32)
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 17.59 FLXA0FRTXRQi Register Contents

Bit Position	Bit Name	Function
31 to 0	TXRo	Transmission Request Flag o

(1) FLXA0FRTXRQi.TXRo (o = (i-1)*32 to (i*32-1))

Transmission Request Flag o

If the flag is set to ‘1’, the respective message buffer is ready for transmission or transmission of this message buffer is in progress.

In single-shot mode the flags are reset to ‘0’ after transmission has completed.

This bit is cleared by the CHI command CLEAR_RAMS.

17.2.9.6 FLXA0FRNDATi — FlexRay New Data Register i (i = 1 to 4)

The four registers reflect the state of the ND flags of all configured message buffers. ND flags belonging to transmit buffers have no meaning. If the number of configured message buffers is less than 128, the remaining ND flags have no meaning.

Access: This register can be read only in 8-, 16-, or 32-bit units.

Address: ERAY + 0330_H to ERAY + $033C_H$ (ERAY + 0330_H + (i-1)*4)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	NDn (n = (i - 1) × 32 + 31)	NDn (n = (i - 1) × 32 + 30)	NDn (n = (i - 1) × 32 + 29)	NDn (n = (i - 1) × 32 + 28)	NDn (n = (i - 1) × 32 + 27)	NDn (n = (i - 1) × 32 + 26)	NDn (n = (i - 1) × 32 + 25)	NDn (n = (i - 1) × 32 + 24)	NDn (n = (i - 1) × 32 + 23)	NDn (n = (i - 1) × 32 + 22)	NDn (n = (i - 1) × 32 + 21)	NDn (n = (i - 1) × 32 + 20)	NDn (n = (i - 1) × 32 + 19)	NDn (n = (i - 1) × 32 + 18)	NDn (n = (i - 1) × 32 + 17)	NDn (n = (i - 1) × 32 + 16)
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	NDn (n = (i - 1) × 32 + 15)	NDn (n = (i - 1) × 32 + 14)	NDn (n = (i - 1) × 32 + 13)	NDn (n = (i - 1) × 32 + 12)	NDn (n = (i - 1) × 32 + 11)	NDn (n = (i - 1) × 32 + 10)	NDn (n = (i - 1) × 32 + 9)	NDn (n = (i - 1) × 32 + 8)	NDn (n = (i - 1) × 32 + 7)	NDn (n = (i - 1) × 32 + 6)	NDn (n = (i - 1) × 32 + 5)	NDn (n = (i - 1) × 32 + 4)	NDn (n = (i - 1) × 32 + 3)	NDn (n = (i - 1) × 32 + 2)	NDn (n = (i - 1) × 32 + 1)	NDn (n = (i - 1) × 32)
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 17.60 FLXA0FRNDATi Register Contents

Bit Position	Bit Name	Function
31 to 0	NDn	New Data Flag n

(1) FLXA0FRNDATi.NDn (n = (i-1)*32 to (i*32-1))

New Data Flag n

The flags are set to ‘1’ when a valid received data frame matches the message buffer’s filter configuration, independent of the payload length received or the payload length configured for that message buffer.

The flags are not set to ‘1’ after reception of null frames except for message buffers belonging to the receive FIFO.

An ND flag is reset to ‘0’ when the header section of the corresponding message buffer is reconfigured or when the data section has been transferred to the output buffer.

Reset when leaving CONFIG state or when entering STARTUP state.

This bit is cleared by the CHI command CLEAR_RAMS.

17.2.9.7 FLXA0FRMBSCi — FlexRay Message Buffer Status Changed Register i (i = 1 to 4)

The four registers reflect the state of the MBC flags of all configured message buffers. If the number of configured message buffers is less than 128, the remaining MBC flags have no meaning.

Access: This register can be read only in 8-, 16-, or 32-bit units.

Address: ERAY + 0340_H to ERAY + $034C_H$ (ERAY + 0340_H + (i-1)*4)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MBCm (m = (i - 1) × 32 + 31)	MBCm (m = (i - 1) × 32 + 30)	MBCm (m = (i - 1) × 32 + 29)	MBCm (m = (i - 1) × 32 + 28)	MBCm (m = (i - 1) × 32 + 27)	MBCm (m = (i - 1) × 32 + 26)	MBCm (m = (i - 1) × 32 + 25)	MBCm (m = (i - 1) × 32 + 24)	MBCm (m = (i - 1) × 32 + 23)	MBCm (m = (i - 1) × 32 + 22)	MBCm (m = (i - 1) × 32 + 21)	MBCm (m = (i - 1) × 32 + 20)	MBCm (m = (i - 1) × 32 + 19)	MBCm (m = (i - 1) × 32 + 18)	MBCm (m = (i - 1) × 32 + 17)	MBCm (m = (i - 1) × 32 + 16)
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MBCm (m = (i - 1) × 32 + 15)	MBCm (m = (i - 1) × 32 + 14)	MBCm (m = (i - 1) × 32 + 13)	MBCm (m = (i - 1) × 32 + 12)	MBCm (m = (i - 1) × 32 + 11)	MBCm (m = (i - 1) × 32 + 10)	MBCm (m = (i - 1) × 32 + 9)	MBCm (m = (i - 1) × 32 + 8)	MBCm (m = (i - 1) × 32 + 7)	MBCm (m = (i - 1) × 32 + 6)	MBCm (m = (i - 1) × 32 + 5)	MBCm (m = (i - 1) × 32 + 4)	MBCm (m = (i - 1) × 32 + 3)	MBCm (m = (i - 1) × 32 + 2)	MBCm (m = (i - 1) × 32 + 1)	MBCm (m = (i - 1) × 32)
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 17.61 FLXA0FRMBSCi (i = 1 to 4) Register Contents

Bit Position	Bit Name	Function
31 to 0	MBCm	Message Buffer Status Changed Flag m

(1) FLXA0FRMBSCi.MBCm (m = (i-1)*32 to (i*32-1))

Message Buffer Status Changed Flag m

Indicates whether the message handler has changed one of the status flags VFRA, VFRB, SEOA, SEOB, CEOA, CEOB, SVOA, SVOB, TCIA, TCIB, ESA, ESB, MLST, FTA, FTB in the header section (see **Section 17.2.11.5, FLXA0FRMBS — FlexRay Message Buffer Status Register** and **Section 17.3.13.1, Header Partition** of the respective message buffer.

An MBC flag is reset to '0' when the header section of the corresponding message buffer is reconfigured or when it has been transferred to the output buffer.

Reset when leaving CONFIG state or when entering STARTUP state.

This bit is cleared by the CHI command CLEAR_RAMS.

17.2.10 Input Buffer

The input buffer (IBF) has a double structure of IBF host and IBF shadow. While the host can write to IBF host, the transfer to the message RAM is done from IBF shadow. The input buffer holds the header and data sections to be transferred to the selected message buffer in the message RAM. It is used to configure the message buffers in the message RAM and to update the data sections of transmit buffers.

When updating the header section of a message buffer in the message RAM from the input buffer, the message buffer status as described in **Section 17.2.11.5, FLXA0FRMBS — FlexRay Message Buffer Status Register** is automatically reset to 0.

The header sections of message buffers belonging to the receive FIFO can only be configured or reconfigured when the CC is in DEFAULT_CONFIG or CONFIG state. For those message buffers, only the payload length and the data pointer can be configured via FLXA0FRWRHS2.PLC and FLXA0FRWRHS3.DP. All information required for acceptance filtering is taken from the FIFO rejection filter and the FIFO rejection filter mask registers.

The data transfer between input buffer (IBF) and message RAM is described in detail in **Section 17.3.12.2, Host Access to Message RAM**.

These registers cannot be written when the input data transfer function shown in **Section 17.3.16.1, Input Data Transfer** is used and the FLXA0FRITS.ITS bit is 1.

17.2.10.1 FLXA0FRWRDSn — FlexRay Write Data Section Register n (n = 1 to 64)

This register holds the data words to be transferred to the data section of the specified message buffer. The number of data words (DWn) written to the message RAM is defined by the payload length configured in the FLXA0FRWRHS2.PLC bit.

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: ERAY + 0400_H to ERAY + $04FC_H$(ERAY + 0400_H + (n-1)*4)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MD[31:16]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MD[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 17.62 FLXA0FRWRDSn Register Contents

Bit Position	Bit Name	Function
31 to 0	MD[31:0]	Message Data

(1) FLXA0FRWRDSn.MD

Message Data Bits

For information about the byte alignment of the message data in this register see **Section 17.3.17, Byte Alignment**.

NOTE

1. **In case FLXA0FRWRHS2.PLC specifies an odd payload length, the remaining message data bytes are unused.**
2. **When writing to FLXA0FRWRDSn, each 32-bit word has to be filled up by access in a single 32- bit unit, in two consecutive 16-bit units, or in four consecutive 8-bit units before transfer from the input buffer to the message RAM will start. If not all bytes of a 32- bit word have been written by the host (8- or 16-bit access only), FLXA0FRWRDSn holds partly undefined data.**
   Reset by the CHI command CLEAR_RAMS.

17.2.10.2 FLXA0FRWRHS1 — FlexRay Write Header Section Register 1

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: ERAY + 0500_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	MBI	TXM	PPIT	CFG	CH[1:0]		—	CYC[6:0]						
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	FID[10:0]										
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 17.63 FLXA0FRWRHS1 Register Contents

Bit Position	Bit Name	Function
31, 30	—	Reserved These bits are always read as 0. When writing, always write 0.
29	MBI	Message Buffer Interrupt 0: The corresponding message buffer interrupt is disabled 1: The corresponding message buffer interrupt is enabled
28	TXM	Transmission Mode 0: Continuous mode 1: Single-shot mode
27	PPIT	Payload Preamble Indicator Transmit 0: Payload Preamble Indicator is set to ‘0’ 1: Payload Preamble Indicator is set to ‘1’
26	CFG	Message Buffer Direction Configuration 0: The corresponding buffer is configured as receive buffer 1: The corresponding buffer is configured as transmit buffer
25,24	CH[1:0]	Channel Filter Control
23	—	Reserved This bit is always read as 0. When writing, always write 0.
22 to 16	CYC[6:0]	Cycle Code
15 to 11	—	Reserved These bits are always read as 0. When writing, always write 0.
10 to 0	FID[10:0]	Frame ID

(1) FLXA0FRWRHS1.MBI

Message Buffer Interrupt Enable

This bit enables the receive/transmit interrupt for the corresponding message buffer.

After a dedicated receive buffer has been updated by the message handler, flag FLXA0FRSIR.RXI and/or FLXA0FRSIR.MBSI are set to '1'. After a transmission has completed, flag FLXA0FRSIR.TXI is set to '1'.

(2) FLXA0FRWRHS1.TXM

Transmission Mode Setting

This bit selects transmit mode of the corresponding message buffer. For transmit mode, see **Section 17.3.9.3, Transmit Buffers**.

(3) FLXA0FRWRHS1.PPIT

Payload Preamble Indicator Transmit

This bit is used to control the state of the Payload Preamble Indicator in transmit frames in transmit frames of the corresponding message buffer.

If the bit is set to '1' in a static message buffer, the respective message buffer holds network management information.

If the bit is set to '1' in a dynamic message buffer, the first two bytes of the payload segment may be used for message ID filtering by the receiver. Message ID filtering of received FlexRay frames is not supported by the FlexRay module, but can be done by the host.

(4) FLXA0FRWRHS1.CFG

Message Buffer Direction Configuration

This bit is used to configure the corresponding buffer as transmit buffer or as receive buffer. For message buffers belonging to the receive FIFO the bit is disabled.

If an unused area of at least 32 bits is not allocated at the start of the data partition, set this bit to 1 to select allocation of the data section of the message buffer as a transmission buffer immediately following (after the last buffer of) the header partition.

(5) FLXA0FRWRHS1.CH

Channel Filter Control

The 2-bit channel filtering field associated with each buffer serves as a filter for receive buffers, and as a control field for transmit buffers.

CH[1:0]	Transmit Buffer transmit frame on	Receive Buffer store frame received from
00	Transmission prohibited	ignore frame
01	channel A	channel A
10	channel B	channel B
11	both channels (static segment only)	channel A or B (store first semantically valid frame; static segment only)

NOTE

If a message buffer is configured for the dynamic segment and both bits of the channel filtering field are set to '1', no transmission nor reception of frames is performed. (same function as CH = "00")

(6) FLXA0FRWRHS1.CYC

Cycle Code

The 7-bit cycle code determines the cycle set used for cycle counter filtering.

For details about the configuration of the cycle code **Section 17.3.8.2, Cycle Counter Filtering**.

(7) FLXA0FRWRHS1.FID

Frame ID

Frame ID of the selected message buffer. The frame ID defines the slot number for transmission/ reception of the respective message.

Message buffers with frame ID = ‘0’ are considered as invalid.

17.2.10.3 FLXA0FRWRHS2 — FlexRay Write Header Section Register 2

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: ERAY + 0504_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	PLC[6:0]						
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	CRC[10:0]										
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 17.64 FLXA0FRWRHS2 Register Contents

Bit Position	Bit Name	Function
31 to 23	—	Reserved These bits are always read as 0. When writing, always write 0.
22 to 16	PLC[6:0]	Payload Length Configured
15 to 11	—	Reserved These bits are always read as 0. When writing, always write 0.
10 to 0	CRC[10:0]	Header CRC (vRF!Header!HeaderCRC) Receive Buffer: Configuration is not required Transmit Buffer: Header CRC is configured

(1) FLXA0FRWRHS2.PLC

Payload Length Configured

Length of data section (number of 2-byte words) as configured by the host.

During static segment the static frame payload length as configured by FLXA0FRMHDC.SFDL defines the payload length for all static frames. If the payload length configured by FLXA0FRWRHS2.PLC is shorter than this value padding bytes are inserted to ensure that frames have proper physical length. The padding pattern is "0000_H" (see **Section 17.3.9.3, Transmit Buffers**).

(2) FLXA0FRWRHS2.CRC

Header CRC (vRF!Header!HeaderCRC)

Setting of the receive buffer is not required.

Transmitting of the message buffer needs the header CRC calculation and setting.

For calculation of the header CRC, the payload length of the frame to be sent has to be considered. In static segment the payload length of all frames is configured by FLXA0FRMHDC.SFDL.

17.2.10.4 FLXA0FRWRHS3 — FlexRay Write Header Section Register 3

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: ERAY + 0508_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	DP[10:0]										
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 17.65 FLXA0FRWRHS3 Register Contents

Bit Position	Bit Name	Function
31 to 11	—	Reserved These bits are always read as 0. When writing, always write 0.
10 to 0	DP[10:0]	Data Pointer

(1) FLXA0FRWRHS3.DP

Data Pointer

Configures the pointer to the first 32-bit word of the data section of the addressed message buffer in the message RAM.

17.2.10.5 FLXA0FRIBCM — FlexRay Input Buffer Command Mask Register

Configures how the message buffer in the message RAM selected by register FLXA0FRIBCR is updated. When IBF host and IBF shadow are swapped, also mask bits FLXA0FRIBCM.LHSH, FLXA0FRIBCM.LDSH, and FLXA0FRIBCM.STXRH are swapped with bits FLXA0FRIBCM.LHSS, FLXA0FRIBCM.LDSS, and FLXA0FRIBCM.STXRS.

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: ERAY + 0510_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	STXRS	LDSS	LHSS
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	STXRH	LDSH	LHSH
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Table 17.66 FLXA0FRIBCM Register Contents

Bit Position	Bit Name	Function
31 to 19	—	Reserved These bits are always read as 0. When writing, always write 0.
18	STXRS	Transmission Request Shadow Set Flag 0: Reset TXR flag 1: Set TXR flag, transmit buffer released for transmission (operation ongoing or finished)
17	LDSS	Data Section Shadow Load Flag 0: Data section is not updated 1: Data section is transferred (transfer ongoing or finished)
16	LHSS	Header Section Shadow Load Flag 0: Header section is not updated 1: Header is transferred (transfer ongoing or finished)
15 to 3	—	Reserved These bits are always read as 0. When writing, always write 0.
2	STXRH	Transmission Request Host Set Flag 0: Reset TXR flag 1: Set TXR flag, transmit buffer released for transmission
1	LDSH	Data Section Host Load Flag 0: Data section is not updated 1: Data section is transferred
0	LHSH	Header Section Host Load Flag 0: Header section is not updated 1: Header is transferred

(1) FLXA0FRIBCM.STXRS

Transmission Request Shadow Set Flag

(2) FLXA0FRIBCM.LDSS

Data Section Shadow Load Flag

(3) FLXA0FRIBCM.LHSS

Header Section Shadow Load Flag

(4) FLXA0FRIBCM.STXRH

Transmission Request Host Set Flag

If this bit is set to '1', the TXR flag for the selected message buffer is set in the FLXA0FRTXRQ1/2/3/4 registers to release the message buffer for transmission. In single-shot mode the flag is cleared after transmission has completed.

TXR is enabled for transmit buffers only.

(5) FLXA0FRIBCM.LDSH

Data Section Host Load Flag

(6) FLXA0FRIBCM.LHSH

Header Section Host Load Flag

17.2.10.6 FLXA0FRIBCR — FlexRay Input Buffer Command Request Register

When the host writes the number of the target message buffer in the message RAM to FLXA0FRIBCR.IBRH,

IBF host and IBF shadow are switched. In addition the message buffer numbers stored under FLXA0FRIBCR.IBRH and FLXA0FRIBCR.IBRS are also switched (see **Section 17.3.12.2 (1), Data Transfer from Input Buffer to Message RAM**).

With this write operation, the FLXA0FRIBCR.IBSYS is set to '1'. The message handler then starts to transfer the contents of IBF shadow to the message buffer in the message RAM selected by FLXA0FRIBCR.IBRS.

After the transfer between IBF shadow and the message RAM has completed, FLXA0FRIBCR.IBSYS is set back to '0' and the next transfer to the message RAM may be started by the host by writing the respective target message buffer number to FLXA0FRIBCR.IBRH.

If a write access to FLXA0FRIBCR.IBRH occurs while FLXA0FRIBCR.IBSYS is '1', FLXA0FRIBCR.IBSYH is set to '1'. After completion of the ongoing data transfer from IBF Shadow to the message RAM, IBF host and IBF shadow are switched, FLXA0FRIBCR.IBSYH is reset to '0'. FLXA0FRIBCR.IBSYS remains set to '1', and the next transfer to the message RAM is started. In addition the message buffer numbers stored under FLXA0FRIBCR.IBRH and FLXA0FRIBCR.IBRS are also switched.

Any write access to an input buffer register while both FLXA0FRIBCR.IBSYS and FLXA0FRIBCR.IBSYH are set to '1' will cause the error flag FLXA0FREIR.IIBA to be set to '1'. In this case, this write access has no effect and the input buffer will not be changed.

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: ERAY + 0514_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	IBSYS	—	—	—	—	—	—	—	—	IBRS[6:0]						
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IBSYH	—	—	—	—	—	—	—	—	IBRH[6:0]						
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 17.67 FLXA0FRIBCR Register Contents (1/2)

Bit Position	Bit Name	Function
31	IBSYS	Input Buffer Busy Shadow Flag 0: Transfer between IBF shadow and message RAM completed 1: Transfer between IBF shadow and message RAM in progress
30 to 23	—	Reserved These bits are always read as 0. When writing, always write 0.
22 to 16	IBRS[6:0]	Input Buffer Request ShadowFlags
15	IBSYH	Input Buffer Busy Host Flag 0: No request pending 1: Request while transfer between IBF shadow and message RAM in progress

Table 17.67 FLXA0FRIBCR Register Contents (2/2)

Bit Position	Bit Name	Function
14 to 7	—	Reserved These bits are always read as 0. When writing, always write 0.
6 to 0	IBRH[6:0]	Input Buffer Request Host

(1) FLXA0FRIBCR.IBSYS

Input Buffer Busy Shadow Flag

Set to ‘1’ after writing FLXA0FRIBCR.IBRH.

This bit indicates transmitting between the IBF shadow and the message RAM is ongoing.

When the transfer between IBF shadow and the message RAM has completed, FLXA0FRIBCR.IBSYS is set back to ‘0’.

(2) FLXA0FRIBCR.IBRS

Input Buffer Request Shadow Flags

Number of the target message buffer actually updated or lately updated.

(3) FLXA0FRIBCR.IBSYH

Input Buffer Busy Host Flag

Set to ‘1’ by writing FLXA0FRIBCR.IBRH while FLXA0FRIBCR.IBSYS is still ‘1’.

This bit indicates transmitting between the IBF shadow and the message RAM is ongoing.

After the ongoing transfer between IBF shadow and the message RAM has completed, the FLXA0FRIBCR.IBSYH is set back to ‘0’.

(4) FLXA0FRIBCR.IBRH

Input Buffer Request Host

Selects the target message buffer in the message RAM for data transfer from input buffer.

17.2.11 Output Buffer

The output buffer has a double structure of output buffer host and output buffer shadow and is used to read out message buffers from the message RAM. While the host can read from OBF host, the message handler transfers the selected message buffer from message RAM to output buffer shadow. The data transfer between message RAM and OBF is described in **Section 17.3.12.2 (2), Data Transfer from Message RAM to Output Buffer**.

These registers cannot be written when the output data transfer function shown in **Section 17.3.16.2, Output Data Transfer**, in output data transfer is used and the FLXA0FROTS.OTS bit is 1.

17.2.11.1 FLXA0FRRDDSn — FlexRay Read Data Section Register n (n = 1 to 64)

These registers hold the data words read from the data section of the specified message buffer. The number of data word read from the message RAM is defined by the payload length configured in the FLXA0FRRDHS2.PLC bit.

Access: This register can be read in 8-, 16-, or 32-bit units.

Address: ERAY + 0600_H to ERAY + $06FC_H$ (ERAY + 0600_H + (n-1)*4)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MD[31:16]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MD[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 17.68 FLXA0FRRDDSn Register Contents

Bit Position	Bit Name	Function
31 to 0	MD[31:0]	Message Data Flags

(1) FLXA0FRRDDSn.MD

Message Data Flags

For information about the byte alignment of the data words in this register, see **Section 17.3.17, Byte Alignment**.

NOTE

In case FLXA0FRWRHS2.PLC specifies an odd payload length, the remaining message data bytes are unused.

Reset by the CHI command CLEAR_RAMS.

17.2.11.2 FLXA0FRRDHS1 — FlexRay Read Header Section Register 1

Access: This register can be read in 8-, 16-, or 32-bit units.

Address: ERAY + 0700_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	MBI	TXM	PPIT	CFG	CH[1:0]		—	CYC[6:0]						
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	FID[10:0]										
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 17.69 FLXA0FRRDHS1 Register Contents

Bit Position	Bit Name	Function
31, 30	—	Reserved These bits are always read as 0. When writing, always write 0.
29	MBI	Message Buffer Interrupt Enable Flag
28	TXM	Transmission Mode Flag
27	PPIT	Payload Preamble Indicator Transmit Flag
26	CFG	Message Buffer Direction Configuration Bit Flag
25, 24	CH[1:0]	Channel Filter Control Flag
23	—	Reserved This bit is always read as 0. When writing, always write 0.
22 to 16	CYC[6:0]	Cycle Code
15 to 11	—	Reserved These bits are always read as 0. When writing, always write 0.
10 to 0	FID[10:0]	Frame ID

(1) FLXA0FRRDHS1.MBI

Message Buffer Interrupt Enable Flag

Values as configured by the host via FLXA0FRWRHS1.MBI.

In case that the message buffer read from the message RAM belongs to the receive FIFO, this bit is set to ‘0’.

(2) FLXA0FRRDHS1.TXM

Transmission Mode Flag

Values as configured by the host via FLXA0FRWRHS1.TXM.

In case that the message buffer read from the message RAM belongs to the receive FIFO, this bit is set to ‘0’.

(3) FLXA0FRRDHS1.PPIT

Payload Preamble Indicator Transmit Flag

Values as configured by the host via FLXA0FRWRHS1.PPIT.

In case that the message buffer read from the message RAM belongs to the receive FIFO, this bit is set to ‘0’.

(4) FLXA0FRRDHS1.CFG

Message Buffer Direction Configuration Flag

Values as configured by the host via FLXA0FRWRHS1.CFG.

In case that the message buffer read from the message RAM belongs to the receive FIFO, this bit is set to ‘0’.

(5) FLXA0FRRDHS1.CH

Channel Filter Control Flag

Values as configured by the host via FLXA0FRWRHS1.CH.

In case that the message buffer read from the message RAM belongs to the receive FIFO, these bits are set to ‘0’.

(6) FLXA0FRRDHS1.CYC

Cycle Code

Values as configured by the host via FLXA0FRWRHS1.CYC.

In case that the message buffer read from the message RAM belongs to the receive FIFO, these bits are set to ‘0’.

(7) FLXA0FRRDHS1.FID

Frame ID

Values as configured by the host via FLXA0FRWRHS1.FID

In case that the message buffer read from the message RAM belongs to the receive FIFO, these bits are holding the received frame ID.

17.2.11.3 FLXA0FRRDHS2 — FlexRay Read Header Section Register 2

Access: This register can be read in 8-, 16-, or 32-bit units.

Address: ERAY + 0704_H

Value after reset: $0000\ 0000_H$

CAUTION

For message buffers belonging to the Static Buffer area or Static and Dynamic Buffer area FLXA0FRWRHS2 is updated from data frames only.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	PLR[6:0]							—	PLC[6:0]						
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	CRC[10:0]										
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 17.70 FLXA0FRRDHS2 Register Contents

Bit Position	Bit Name	Function
31	—	Reserved This bit is always read as 0. When writing, always write 0.
30 to 24	PLR[6:0]	Receive Frame Payload Length Flags (vRF!Header!Length)
23	—	Reserved This bit is always read as 0. When writing, always write 0.
22 to 16	PLC[6:0]	Configured Payload Length Flags
15 to 11	—	Reserved These bits are always read as 0. When writing, always write 0.
10 to 0	CRC[10:0]	Header CRC Flags (vRF!Header!HeaderCRC)

(1) FLXA0FRRDHS2.PLR

Receive Frame Payload Length Flags (vRF!Header!Length)

Payload length (vRF!Header!Length) value updated from received data frames (exception: To be updated even through null frames are received in the message buffer specified to the receive FIFO).

(2) FLXA0FRRDHS2.PLC

Configured Payload Length Flags

Length of data section (number of 2-byte words) as configured by the host.

(3) FLXA0FRRDHS2.CRC

Header CRC Flags (vRF!Header!HeaderCRC)

Receive Buffer: Header CRC (vRF!Header!HeaderCRC) updated from received data frames

Transmit Buffer: Header CRC configured by the host

(4) Data storage

When a message is stored into a message buffer the following processes with respect to payload length received and payload length configured are implemented:

FLXA0FRRDHS2.PLR > FLXA0FRRDHS2.PLC:

The payload data stored in the message buffer is truncated to the payload length configured if FLXA0FRRDHS2.PLC even or else truncated to FLXA0FRRDHS2.PLC + 1.

FLXA0FRRDHS2.PLR <= FLXA0FRRDHS2.PLC:

The received payload data is stored into the message buffers data section. The remaining data bytes of the data section as configured by FLXA0FRRDHS2.PLC are filled with undefined value.

FLXA0FRRDHS2.PLR = zero:

The message buffer's data section is filled with undefined value.

FLXA0FRRDHS2.PLC = zero:

No data section is configured in the message buffer. No data is stored into the message buffer's data section.

NOTE

1. **The message RAM is organized in 4-byte words. When received data is stored into a message buffer's data section, the number of 2-byte data words written into the message buffer is FLXA0FRRDHS2.PLC rounded to the next even value.**
2. **FLXA0FRRDHS2.PLC should be configured identical for all message buffers belonging to the receive FIFO.**
   For message buffers belonging to the Static Buffer area or Static and Dynamic Buffer area FLXA0FRWRHS2 is updated from data frames only.

17.2.11.4 FLXA0FRRDHS3 — FlexRay Read Header Section Register 3

NOTE

For message buffers belonging to the Static Buffer area or Static and Dynamic Buffer area FLXA0FRWRHS3 is updated from data frames only.

Access: This register can be read in 8-, 16-, or 32-bit units.

Address: ERAY + 0708_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	RES	PPI	NFI	SYN	SFI	RCI	—	—	RCC[5:0]					
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	DP[10:0]										
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 17.71 FLXA0FRRDHS3 Register Contents

Bit Position	Bit Name	Function
31, 30	—	Reserved These bits are always read as 0. When writing, always write 0.
29	RES	Reserved Bit Indicator Flag (vRF!Header!Reserved)
28	PPI	Payload Preamble Indicator Flag (vRF!Header!PPIndicator)
27	NFI	Null Frame Indicator Flag (vRF!Header!NFIndicator) 0 = Up to now no data frame has been stored into the respective message buffer 1 = At least one data frame has been stored into the respective message buffer
26	SYN	Sync Frame Indicator Flag (vRF!Header!SyFIndicator) 0: The received frame is not a sync frame 1: The received frame is a sync frame
25	SFI	Startup Frame Indicator Flag (vRF!Header!SuFIndicator) 0: The received frame is not a startup frame 1: The received frame is a startup frame
24	RCI	Received on Channel Indicator Flag (vSS!Channel) 0: Frame received on channel B 1: Frame received on channel A
23, 22	—	Reserved These bits are always read as 0. When writing, always write 0.
21 to 16	RCC[5:0]	Receive Cycle Count (vRF!Header!CycleCount)
15 to 11	—	Reserved These bits are always read as 0. When writing, always write 0.
10 to 0	DP[10:0]	Data Pointer Flags

(1) FLXA0FRRDHS3.RES

Reserved Bit Indicator Flag (vRF!Header!Reserved)

Reflects the value of the received reserved bit. The reserved bit is transmitted as ‘0’.

(2) FLXA0FRRDHS3.PPI

Payload Preamble Indicator Flag (vRF!Header!PPIndicator)

The payload preamble indicator defines whether a network management vector or message ID is contained in the payload segment of the received frame.

0 = The payload segment of the received frame does not contain a network management vector nor a message ID

1 = Static segment: contains network management vector is included in the first part of the payload
Dynamic segment: contains message ID is included in the first part of the payload

(3) FLXA0FRRDHS3.NFI

Null Frame Indicator Flag (vRF!Header!NFIndicator)

Is set to ‘1’ after storage of the first received data frame.

(4) FLXA0FRRDHS3.SYN

Sync Frame Indicator Flag (vRF!Header!SyFIndicator)

A sync frame is indicated by the sync frame indicator.

(5) FLXA0FRRDHS3.SFI

Startup Frame Indicator Flag (vRF!Header!SuFIndicator)

A startup frame is indicated by the startup frame indicator.

(6) FLXA0FRRDHS3.RCI

Received on Channel Indicator Flag (vSS!Channel)

Indicates the channel from which the received data frame was taken to update the respective receive buffer.

(7) FLXA0FRRDHS3.RCC

Receive Cycle Count (vRF!Header!CycleCount)

Cycle counter value updated from received data frame is read.

(8) FLXA0FRRDHS3.DP

Data Pointer Flags

This flag indicates the position of the first 32-bit word of the data section of the addressed message buffer in the message RAM.

The bit value is the same as that set in the FLXA0FRWRHS3.DP bit.

17.2.11.5 FLXA0FRMBS — FlexRay Message Buffer Status Register

The message buffer status is updated by the CC with respect to the assigned channel(s) latest at the end of the slot following the slot assigned to the message buffer.

The flags are updated only when the CC is in NORMAL_ACTIVE or NORMAL_PASSIVE state.

If only one channel (A or B) is assigned to a message buffer, the channel-specific status flags of the other channel are written to zero. If both channels are assigned to a message buffer, the channel-specific status flags of both channels are updated.

The message buffer status is updated only when the slot counter reached the configured frame ID and when the cycle counter filter matched. When the host updates a message buffer via input buffer, all FLXA0FRMBS flags are reset to zero independent of which FLXA0FRIBCM bits are set or not.

For details about receive/transmit filtering see **Section 17.3.8, Filtering and Masking**, **Section 17.3.9, Transmit Process** and **Section 17.3.10, Receive Process**.

Whenever the message handler changes one of the flags FLXA0FRMBS.VFRA, FLXA0FRMBS.VFRB, FLXA0FRMBS.SEOA, FLXA0FRMBS.SEOB, FLXA0FRMBS.CEOA, FLXA0FRMBS.CEOB, FLXA0FRMBS.SVOA, FLXA0FRMBS.SVOB, FLXA0FRMBS.TCIA, FLXA0FRMBS.TCIB, FLXA0FRMBS.ESA, FLXA0FRMBS.ESB, FLXA0FRMBS.MLST, FLXA0FRMBS.FTA, FLXA0FRMBS.FTB the respective message buffer's MBC flag in registers FLXA0FRMBSC1/2/3/4 is set.

Access: This register can be read in 8-, 16-, or 32-bit units.

Address: ERAY + $070C_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	RESS	PPIS	NFIS	SYNS	SFIS	RCIS	—	—	CCS[5:0]					
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	FTB	FTA	—	MLST	ESB	ESA	TCIB	TCIA	SVOB	SVOA	CEOB	CEOA	SEOB	SEOA	VFRB	VFRA
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 17.72 FLXA0FRMBS Register Contents (1/2)

Bit Position	Bit Name	Function
31, 30	—	Reserved These bits are always read as 0. When writing, always write 0.
29	RESS	Reserved Bit Status Flag (vRF!Header!Reserved)
28	PPIS	Payload Preamble Indicator Status Flag (vRF!Header!PPIndicator) 0: PPI indicator set to ‘0’ 1: PPI indicator set to ‘1’
27	NFIS	Null Frame Indicator Status Flag (vRF!Header!NFIndicator) 0: Received frame is a null frame 1: Received frame is not a null frame
26	SYNS	Sync Frame Indicator Status Flag (vRF!Header!SyFIndicator) 0: No sync frame received 1: The received frame is a sync frame

Table 17.72 FLXA0FRMBS Register Contents (2/2)

Bit Position	Bit Name	Function
25	SFIS	Startup Frame Indicator Status Flag (vRF!Header!SuFIndicator) 0: No startup frame received 1: The received frame is a startup frame
24	RCIS	Received Channel Indicator Status Flag (vSS!Channel) 0: Frame received on channel B 1: Frame received on channel A
23, 22	—	Reserved These bits are always read as 0. When writing, always write 0.
21 to 16	CCS[5:0]	Cycle Count Status Flags
15	FTB	Frame Transmitted on Channel B Flag 0: No data frame transmitted on channel B 1: Data frame transmitted on channel B
14	FTA	Frame Transmitted on Channel A Flag 0: No data frame transmitted on channel A 1: Data frame transmitted on channel A
13	—	Reserved This bit is always read as 0. When writing, always write 0.
12	MLST	Message Lost Flag 0: No message lost 1: Unprocessed message was overwritten
11	ESB	Empty Slot Channel B Flag 0: Bus activity detected in the assigned slot on channel B 1: No bus activity detected in the assigned slot on channel B
10	ESA	Empty Slot Channel A Flag 0: Bus activity detected in the assigned slot on channel A 1: No bus activity detected in the assigned slot on channel A
9	TCIB	Transmission Conflict Indication Channel B Flag (vSS!TxConflictB) 0: No transmission conflict occurred on channel B 1: Transmission conflict occurred on channel B
8	TCIA	Transmission Conflict Indication Channel A Flag (vSS!TxConflictA) 0: No transmission conflict occurred on channel A 1: Transmission conflict occurred on channel A
7	SVOB	Slot Boundary Violation Observed on Channel B Flag (vSS!BViolationB) 0: No slot boundary violation observed on channel B 1: Slot boundary violation observed on channel B
6	SVOA	Slot Boundary Violation Observed on Channel A Flag (vSS!BViolationA) 0: No slot boundary violation observed on channel A 1: Slot boundary violation observed on channel A
5	CEOB	Content Error Observed on Channel B Flag (vSS!ContentErrorB) 0: No content error observed on channel B 1: Content error observed on channel B
4	CEOA	Content Error Observed on Channel A Flag(vSS!ContentErrorA) 0: No content error observed on channel A 1: Content error observed on channel A
3	SEOB	Syntax Error Observed on Channel B Flag (vSS!SyntaxErrorB) 0: No syntax error observed on channel B 1: Syntax error observed on channel B
2	SEOA	Syntax Error Observed on Channel A Flag (vSS!SyntaxErrorA) 0: No syntax error observed on channel A 1: Syntax error observed on channel A
1	VFRB	Valid Frame Received on Channel B Flag (vSS!ValidFrameB) 0: No valid frame received on channel B 1: Valid frame received on channel B
0	VFRA	Valid Frame Received on Channel A Flag (vSS!ValidFrameA) 0: No valid frame received on channel A 1: Valid frame received on channel A

(1) FLXA0FRMBS.RESS

Reserved Bit Status Flag (vRF!Header!Reserved)

Reflects the state of the received reserved bit. The reserved bit is transmitted as ‘0’.

For receive buffers (FLXA0FRWRHS1.CFG = ‘0’) this bit is updated from both valid data and null frames. If no valid frame was received, the previous value is maintained. For transmit buffers the flag has no meaning and should be ignored.

(2) FLXA0FRMBS.PPIS

Payload Preamble Indicator Status Flag (vRF!Header!PPIndicator)

The payload preamble indicator defines whether a network management vector or message ID is contained within the payload segment of the received frame.

For receive buffers (FLXA0FRWRHS1.CFG = ‘0’) this bit is updated from both valid data and null frames. If no valid frame was received, the previous value is maintained. For transmit buffers the flag has no meaning and should be ignored.

0 = PPI indicator set to ‘0’
The payload segment of the received frame does not contain a network management vector or a message ID

1 = PPI indicator set to ‘1’
Static segment: contains network management vector at the beginning of the payload
Dynamic segment: contains message ID at the beginning of the payload

(3) FLXA0FRMBS.NFIS

Null Frame Indicator Status Flag (vRF!Header!NFIndicator)

If set to ‘0’ the payload segment of the received frame contains no usable data.

For receive buffers (FLXA0FRWRHS1.CFG = ‘0’) this bit is updated from both valid data and null frames. If no valid frame was received, the previous value is maintained. For transmit buffers the flag has no meaning and should be ignored.

(4) FLXA0FRMBS.SYNS

Sync Frame Indicator Status Flag (vRF!Header!SyFIndicator)

A sync frame is indicated by the sync frame indicator.

For receive buffers (FLXA0FRWRHS1.CFG = ‘0’) this bit is updated from both valid data and null frames. If no valid frame was received, the previous value is maintained. For transmit buffers the flag has no meaning and should be ignored.

(5) FLXA0FRMBS.SFIS

Startup Frame Indicator Status Flag (vRF!Header!SuFIndicator)

The startup frame indicator specifies a startup frame.

For receive buffers (FLXA0FRWRHS1.CFG = ‘0’) this bit is updated from both valid data and null frames. If no valid frame was received, the previous value is maintained. For transmit buffers the flag has no meaning and should be ignored.

(6) FLXA0FRMBS.RCIS

Received Channel Indicator Status Flag (vSS!Channel)

Indicates the channel on which the frame was received.

For receive buffers (FLXA0FRWRHS1.CFG = ‘0’) this bit is updated from both valid data and null frames. If no valid frame was received, the previous value is maintained. For transmit buffers the flag has no meaning and should be ignored.

(7) FLXA0FRMBS.CCS

Cycle Count Status Flag

Actual cycle count at a status update is read.

(8) FLXA0FRMBS.FTB

Frame Transmitted on Channel B Flag

Indicates that this node has transmitted a data frame in the configured slot on channel B.

NOTE

The FlexRay protocol specification requires that FLXA0FRMBS.FTB can only be reset by the host. Therefore the Cycle Count Status FLXA0FRMBS.CCS for this bit is only valid for the cycle where the bit is set to ‘1’.

(9) FLXA0FRMBS.FTA

Frame Transmitted on Channel A Flag

Indicates that this node has transmitted a data frame in the configured slot on channel A.

NOTE

The FlexRay protocol specification requires that FLXA0FRMBS.FTA can only be reset by the host. Therefore the Cycle Count Status FLXA0FRMBS.CCS for this bit is only valid for the cycle where this bit is set to ‘1’.

(10) FLXA0FRMBS.MLST

Message Lost Flag

The flag is set in case the host did not read the message before the message buffer was updated from a received data frame.

Not affected by reception of null frames except for message buffers belonging to the receive FIFO. The flag is reset to ‘0’ by a host write to the message buffer via IBF or when a new message is stored into the message buffer after the message buffers ND flag was reset to ‘0’ by reading out the message buffer via OBF.

(11) FLXA0FRMBS.ESB

Empty Slot Channel B Flag

In an empty slot, the bus is in idle state. This means that any frame transmission is not detected. This state can be checked in static and dynamic slots.

(12) FLXA0FRMBS.ESA

Empty Slot Channel A Flag

In an empty slot, the bus is in idle state. This means that any frame transmission is not detected. This state can be checked in static and dynamic slots.

(13) FLXA0FRMBS.TCIB

Transmission Conflict Indication Channel B Flag (vSS!TxConflictB)

A transmission conflict indication is set to ‘1’ if a transmission conflict has occurred on channel B.

(14) FLXA0FRMBS.TCIA

Transmission Conflict Indication Channel A Flag (vSS!TxConflictA)

A transmission conflict indication is set if a transmission conflict has occurred on channel A.

(15) FLXA0FRMBS.SVOB

Slot Boundary Violation Observed on Channel B Flag (vSS!BViolationB)

A slot boundary violation (channel active at the start or at the end of the assigned slot) was observed on channel B.

(16) FLXA0FRMBS.SVOA

Slot Boundary Violation Observed on Channel A Flag (vSS!BViolationA)

A slot boundary violation (channel active at the start or at the end of the assigned slot) was observed on channel A.

(17) FLXA0FRMBS.CEOB

Content Error Observed on Channel B Flag (vSS!ContentErrorB)

A content error was observed in the assigned slot on channel B.

(18) FLXA0FRMBS.CEOA

Content Error Observed on Channel A Flag (vSS!ContentErrorA)

A content error was observed in the assigned slot on channel A.

(19) FLXA0FRMBS.SEOB

Syntax Error Observed on Channel B Flag (vSS!SyntaxErrorB)

A syntax error was observed in the assigned slot on channel B.

(20) FLXA0FRMBS.SEOA

Syntax Error Observed on Channel A Flag (vSS!SyntaxErrorA)

A syntax error was observed in the assigned slot on channel A.

(21) FLXA0FRMBS.VFRB

Valid Frame Received on Channel B Flag (vSS!ValidFrameB)

This flag is set to “1” when a valid frame was received on channel B.

(22) FLXA0FRMBS.VFRA

Valid Frame Received on Channel A Flag (vSS!ValidFrameA)

This flag is set to “1” when a valid frame was received on channel A.

17.2.11.6 FLXA0FROBCM — FlexRay Output Buffer Command Mask Register

This register configures how the output buffer is updated from the message buffer in the message RAM selected by FLXA0FROBCR.OBRS.

Mask bits FLXA0FROBCM.RDSS and FLXA0FROBCM.RHSS are copied to the register internal storage when a message RAM transfer is requested by FLXA0FROBCR.REQ.

When OBF host and OBF shadow are switched, mask bits FLXA0FROBCM.RDSH and FLXA0FROBCM.RHSH are switched with the register internal storage to keep them attached to the respective output buffer transfer.

The data transfer between output buffer and message RAM is described in detail in **Section 17.3.12.2 (2), Data Transfer from Message RAM to Output Buffer**.

CAUTION

After the transfer of the header section from the message RAM to OBF shadow has completed, the message buffer status changed flag MBC of the selected message buffer in the FLXA0FRMBSC1/2/3/4 registers is cleared. After the transfer of the data section from the message RAM to OBF shadow has completed, the new data flag ND of the selected message buffer in the FLXA0FRNDAT1/2/3/4 registers is cleared.

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: ERAY + 0710_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RDSH	RHSH
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RDSS	RHSS
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Table 17.73 FLXA0FROBCM Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 18	—	Reserved These bits are always read as 0. When writing, always write 0.
17	RDSH	Data Section Host Read Flag 0: Data section is not read 1: Data section selected for transfer from message RAM to output buffer
16	RHSH	Header Section Host Read Flag 0: Header section is not read 1: Header section selected for transfer from message RAM to output buffer
15 to 2	—	Reserved These bits are always read as 0. When writing, always write 0.
1	RDSS	Data Section Shadow Read Flag 0: Data section is not read 1: Data section selected for transfer from message RAM to output buffer

Table 17.73 FLXA0FROBCM Register Contents (2/2)

Bit Position	Bit Name	Function
0	RHSS	Header Section Shadow Read Flag 0: Header section is not read 1: Header section selected for transfer from message RAM to output buffer

(1) FLXA0FROBCM.RDSH

Data Section Host Read Flag

(2) FLXA0FROBCM.RHSH

Header Section Host Read Flag

(3) FLXA0FROBCM.RDSS

Data Section Shadow Read Flag

(4) FLXA0FROBCM.RHSS

Header Section Shadow Read Flag

17.2.11.7 FLXA0FROBCR — FlexRay Output Buffer Command Request Register

After setting bit FLXA0FROBCR.REQ to '1' while FLXA0FROBCR.OBSYS is '0', FLXA0FROBCR.OBSYS is automatically set to '1', FLXA0FROBCR.OBRS is copied to the register internal storage, mask bits FLXA0FROBCM.RDSS and FLXA0FROBCM.RHSS are copied to register FLXA0FROBCM internal storage, and the transfer of the message buffer selected by FLXA0FROBCR.OBRS from the message RAM to OBF shadow is started. When the transfer between the message RAM and OBF shadow has completed, this is signaled by setting FLXA0FROBCM.OBSYS back to '0'.

By setting bit FLXA0FROBCR.VIEW to '1' while FLXA0FROBCR.OBSYS is '0', OBF host and OBF shadow are switched. Additionally mask bits FLXA0FROBCM.RDSH and FLXA0FROBCM.RHSH are switched with the register FLXA0FROBCM internal storage to keep them attached to the respective output buffer transfer. FLXA0FROBCR.OBRH signals the number of the message buffer currently accessible by the host.

If bits FLXA0FROBCR.REQ and FLXA0FROBCR.VIEW are set to '1' with the same write access while FLXA0FROBCR.OBSYS is '0', FLXA0FROBCR.OBSYS is automatically set to '1' and OBF shadow and OBF host are switched. Additionally mask bits FLXA0FROBCM.RDSH and FLXA0FROBCM.RHSH are switched with the registers internal storage to keep them attached to the respective output buffer transfer. Afterwards FLXA0FROBCR.OBRS is copied to the register internal storage, and the transfer of the selected message buffer from the message RAM to OBF shadow is started. While the transfer is ongoing the host can read the message buffer transferred by the previous transfer from OBF host. When the current transfer between message RAM and OBF shadow has completed, this is signaled by setting FLXA0FROBCR.OBSYS back to '0'.

Any write access to FLXA0FROBCR[15:8] while FLXA0FROBCR.OBSYS is set to '1' will cause the error flag FLXA0FREIR.IOBA to be set to '1'. In this case, this write access has no effect and the output buffer will not be changed.

The data transfer between output buffer and message RAM is described in detail in **Section 17.3.12.2 (2), Data Transfer from Message RAM to Output Buffer**.

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: ERAY + 0714_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	OBRH[6:0]						
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OBSYS	—	—	—	—	—	REQ	VIEW	—	OBRS[6:0]						
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 17.74 FLXA0FROBCR Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 23	—	Reserved Flags These bits are always read as 0. When writing, always write 0.
22 to 16	OBRH[6:0]	Output Buffer Request Host

Table 17.74 FLXA0FROBCR Register Contents (2/2)

Bit Position	Bit Name	Function
15	OBSYS	Output Buffer Busy Shadow Flag 0: No transfer in progress 1: Transfer between message RAM and OBF shadow in progress
14 to 10	—	Reserved These bits are always read as 0. When writing, always write 0.
9	REQ	Request Message RAM Transfer 0: No request 1: Transfer to OBF shadow requested
8	VIEW	View Shadow Buffer 0: No action 1: Switch OBF shadow and OBF host
7	—	Reserved This bit is always read as 0. When writing, always write 0.
6 to 0	OBRS[6:0]	Output Buffer Request Shadow

(1) FLXA0FROBCR.OBRH

Output Buffer Request Host Flags

Indicate number of message buffer currently accessible by the host via FLXA0FRRDHS[1...3], FLXA0FRMBS, and FLXA0FRRDDS[1...64].

By writing FLXA0FROBCR.VIEW to ‘1’, OBF host is switched and the transferred message buffer is accessible by the host.

(2) FLXA0FROBCR.OBSYS

Output Buffer Busy Shadow Flag

Sets to ‘1’ after setting bit FLXA0FROBCR.REQ. When the transfer between the message RAM and OBF shadow has completed, FLXA0FROBCR.OBSYS is set back to ‘0’.

(3) FLXA0FROBCR.REQ

Request Message RAM Transfer

Only writeable while FLXA0FROBCR.OBSYS = ‘0’.

Requests transfer of message buffer addressed by FLXA0FROBCR.OBRS from message RAM to OBF shadow.

(4) FLXA0FROBCR.VIEW

View Shadow Buffer

Only writeable while FLXA0FROBCR.OBSYS = ‘0’.

Toggles between OBF shadow and OBF host.

(5) FLXA0FROBCR.OBRS

Output Buffer Request Shadow

Only writeable while FLXA0FROBCR.OBSYS = ‘0’.

Indicate number of source message buffer to be transferred from the message RAM to OBF shadow.

If the number of the first message buffer of the receive FIFO is written to this register, the message buffer addressed by the GET Index (GIDX, see **Section 17.3.11, FIFO Function**) is transferred to OBF shadow.

17.2.12 Data Transfer Control Register

17.2.12.1 FLXA0FRITC — FlexRay Input Transfer Configuration Register

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: ERAY + 0800_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	ITM[6:0]						
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	IQEIE	IQFIE	—	—	—	—	—	—	IQHR	ITE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R/W	R/W

Table 17.75 FLXA0FRITC Register Contents

Bit Position	Bit Name	Function
31 to 23	—	Reserved These bits are always read as 0. When writing, always write 0.
22 to 16	ITM[6:0]	Input Queue Table Max These bits configure the number of entries in the input pointer table the input buffer handler is capable to maintain in the input queue.
15 to 10	—	Reserved These bits are always read as 0. When writing, always write 0.
9	IQEIE	Input Queue Empty Interrupt Enable 0: Disabled 1: Enabled
8	IQFIE	Input Queue Full Interrupt Enable 0: Disabled 1: Enabled
7 to 2	—	Reserved These bits are always read as 0. When writing, always write 0.
1	IQHR	Input Queue Halt Request 0: Input queue run request 1: Input queue halt request
0	ITE	Input Transfer Enable 0: Operation Disable request 1: Operation Enable request

(1) FLXA0FRITC.ITM

Input Queue Table Max

The user can only write to this bit when FLXA0FRITS.ITS is ‘0’.

These bits configure the number of entries in the input pointer table the input buffer handler is capable to maintain in the input queue.

Valid values are 00_H (1 queue entry) to $7F_H$ (128 queue entries).

Note that each entry requires two long words in the input pointer table.

(2) FLXA0FRITC.IQEIE

Input Queue Empty Interrupt Enable

This bit controls the input queue empty interrupt.

0: Disabled
No interrupt will be requested and the input queue empty interrupt will be not generated.

1: Enabled
Input queue empty interrupt will be generated when FLXA0FRITS.IQEIS is ‘1’.

(3) FLXA0FRITC.IQFIE

Input Queue Full Interrupt Enable

This bit controls the input queue full interrupt.

0: Disabled
No interrupt will be requested and the input queue full interrupt will be not generated.

1: Enabled
Input queue full interrupt will be generated when FLXA0FRITS.IQFIS is ‘1’.

(4) FLXA0FRITC.IQHR

Input Queue Halt Request

The IQHR bit should not be set to ‘1’ when FLXA0FRITS.ITS is ‘0’.

This bit requests a halt of the input queue.

The status of the halt request is shown in the FLXA0FRITS.IQH register.

Refer to **Section 17.3.16.1 (5), Halting the input queue** about usage of this bit.

0: Input queue run request
The input queue resumes their operation.

1: Input queue halt request
The input queue gets halted. An active input transfer will be completed but no further transfer request will start.

(5) FLXA0FRITC.ITE

Input Transfer Enable

The user should only set this bit to ‘1’ when FLXA0FRIBCR.IBSYS is ‘0’.

The user should only set this bit to ‘0’ when FLXA0FRITC.IQHR ‘0’. Otherwise committed input transfers get lost.

This bit controls the operation mode of the input transfer queue.

The operation status of the input transfer queue function is shown in FLXA0FRITS.ITS.

Refer to **Section 17.3.16.1 (1), Activation and deactivation** about usage of this bit.

0: Operation Disable request
The input transfer queue gets disabled when it becomes empty.

1: Operation Enable request
The input transfer queue gets enabled. Input data structures are transferred to the FlexRay internal message RAM.

17.2.12.2 FLXA0FROTC — FlexRay Output Transfer Configuration Register

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: ERAY + 0804_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	FTM[4:0]				
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	FWIE	OWIE	FIE	OIE	—	—	—	—	—	—	OTCS	OTE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R/W	R/W

Table 17.76 FLXA0FROTC Register Contents

Bit Position	Bit Name	Function
31 to 21	—	Reserved These bits are always read as 0. When writing, always write 0.
20 to 16	FTM[4:0]	FIFO Table Max Configure the number of FIFO entries the output transfer handler is capable to maintain in the local RAM/global RAM.
15 to 12	—	Reserved These bits are always read as 0. When writing, always write 0.
11	FWIE	FIFO Transfer Warning Interrupt Enable 0: Disabled 1: Enabled
10	OWIE	Output Transfer Warning Interrupt Enable 0: Disabled 1: Enabled
9	FIE	FIFO Transfer Interrupt Enable 0: Disabled 1: Enabled
8	OIE	Output Transfer Interrupt Enable 0: Disabled 1: Enabled
7 to 2	—	Reserved These bits are always read as 0. When writing, always write 0.
1	OTCS	Output Transfer Condition Select 0: New data only mode 1: New data and status changed mode
0	OTE	Output Transfer Enable 0: Operation Disable request 1: Operation Enable request

(1) FLXA0FROTC.FTM

FIFO Table Max

The user can only write to these bits when FLXA0FROTS.OTS is ‘0’.

These bits configure the number of FIFO entries the output transfer handler is capable to maintain in the local RAM/global RAM.

Valid values are 00_H (1 FIFO entry) to $1F_H$ (32 FIFO entries).

(2) FLXA0FROTC.FWIE

FIFO transfer Warning Interrupt Enable

This bit controls the FIFO transfer warning interrupt.

0: Disabled
No interrupt will be requested and the FIFO transfer warning interrupt will be not generated.

1: Enabled
FIFO transfer warning interrupt will be generated when FLXA0FROTS.FWIS is ‘1’.

(3) FLXA0FROTC.OWIE

Output Transfer Warning Interrupt Enable

This bit controls the output transfer warning interrupt.

0: Disabled
No interrupt will be requested and the output transfer warning interrupt will be not generated.

1: Enabled
Output transfer warning interrupt will be generated when FLXA0FROTS.OWIS is ‘1’.

(4) FLXA0FROTC.FIE

FIFO Transfer Interrupt Enable

This bit controls the FIFO transfer interrupt.

0: Disabled
No interrupt will be requested and the FIFO transfer interrupt will be not generated.

1: Enabled
FIFO transfer interrupt will be generated when FLXA0FROTS.FIS is ‘1’.

(5) FLXA0FROTC.OIE

Output Transfer Interrupt Enable

This bit controls the output transfer interrupt.

0: Disabled
No interrupt will be requested and the output transfer interrupt will be not generated.

1: Enabled
Output transfer interrupt will be generated when FLXA0FROTS.OTIS is ‘1’.

(6) FLXA0FROTC.OTCS

Output Transfer Condition Select

The user can only write to this bit when FLXA0FROTS.OTS is '0'.

This bit controls the output transfer condition.

0: New data only mode
The ND bits in the FLXA0FRNDATi registers are used to detect a transfer condition for dedicated receive buffer

1: New data and status changed mode
The ND bits in the FLXA0FRNDATi registers and the MBC bits in the FLXA0FRMBSCi register are used to detect a transfer condition for dedicated transmit and receive buffer

(7) FLXA0FROTC.OTE

Output Transfer Enable

The user should only set this bit to '1' when FLXA0FROBCR.OBSYS is '0'.

This bit controls the operation mode of the output transfer function.

The operation status of the output buffer transfer function is shown in FLXA0FROTS.OTS.

Refer to **Section 17.3.16.2 (1), Activation and deactivation** about usage of this bit.

0: Operation Disable request
The output buffer transfer gets disabled.
An active message buffer transfer will be completed but no further transfer will start.

1: Operation Enable request
The output buffer transfer gets enabled. Message buffers are transferred from the FlexRay internal message RAM to output data structures.

The user should not change the E-Ray message RAM configuration by writing to the FLXA0FRMRC register.

17.2.12.3 FLXA0FRIBA — FlexRay Input Pointer Table Base Address Register

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: ERAY + 0808_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ITA[31:16]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ITA[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Table 17.77 FLXA0FRIBA Register Contents

Bit Position	Bit Name	Function
31 to 0	ITA[31:0]	Input Table Base Address These bits configure the base address of the input pointer table.

(1) FLXA0FRIBA.ITA

Input Table Base Address

The user can only write to these bits when FLXA0FRITS.ITS is ‘0’.

The address should be 32 bit aligned, thus the bits FLXA0FRIBA.ITA[1:0] are always ‘0’.

These bits configure the base address of the input pointer table.

The table is used for the input transfer queue transferring message buffers from the local RAM/global RAM into the FlexRay internal message RAM.

The size of the input queue is configured in FLXA0FRITC.ITM.

Note that each entry requires two long words in the input pointer table.

17.2.12.4 FLXA0FRFBA — FlexRay FIFO Pointer Table Base Address Register

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: ERAY + $080C_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	FTA[31:16]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	FTA[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Table 17.78 FLXA0FRFBA Register Contents

Bit Position	Bit Name	Function
31 to 0	FTA[31:0]	FIFO Pointer Table Base Address These bits configure the base address of the FIFO pointer table.

(1) FLXA0FRFBA.FTA

FIFO Pointer Table Base Address

The user can only write to these bits when FLXA0FROTS.OTS is ‘0’.

The address should be 32 bit aligned, thus the bits FLXA0FRFBA.FTA[1:0] are always ‘0’.

These bits configure the base address of the FIFO pointer table.

The table is used for message buffers transferred from the FlexRay internal FIFO to the local RAM/ global RAM.

The size of the FIFO is configured in FLXA0FROTC.FTM.

17.2.12.5 FLXA0FROBA — FlexRay Output Pointer Table Base Address Register

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: ERAY + 0810_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	OTA[31:16]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OTA[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R

Table 17.79 FLXA0FROBA Register Contents

Bit Position	Bit Name	Function
31 to 0	OTA[31:0]	Output Pointer Table Base Address These bits configure the base address of the output pointer table.

(1) FLXA0FROBA.OTA

Output Pointer Table Base Address

The user can only write to these bits when FLXA0FROTS.OTS is ‘0’.

The address should be 32 bit aligned, thus the bits FLXA0FROBA.OTA[1:0] are always ‘0’.

These bits configure the base address of the output pointer table.

The table is used for message buffers transferred from the FlexRay internal message RAM to the local RAM/global RAM.

The size of the table depends on the utilization of the FlexRay internal message RAM and can have up to 128 entries.

17.2.12.6 FLXA0FRIQC — FlexRay Input Queue Control Register

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: ERAY + 0814_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	IMBNR[6:0]						
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	W	W	W	W	W	W	W

Table 17.80 FLXA0FRIQC Register Contents

Bit Position	Bit Name	Function
31 to 7	—	Reserved These bits are always read as 0. When writing, always write 0.
6 to 0	IMBNR[6:0]	Input Message Buffer Number Message buffer number added to the input queue

(1) FLXA0FRIQC.IMBNR

Input Message Buffer Number

The user can only write to these bits when FLXA0FRITS.IQFP is ‘0’.

The user should not write to this register when FLXA0FRITS.ITS is '0' or when FLXA0FRITC.ITE is ‘0’.

These bits are read as “0”.

This value specifies the message buffer added to the input queue.

The number has to be identical to FLXA0FRWRHS4.IMBNR (see **Section 17.3.16.1 (3), Input pointer table**) of the input pointer table.

The address to the input data structure has to be provided in the input pointer table at the put index (FLXA0FRITS.IPIDX) before writing to this register.

Writing to this register increments the input put index (FLXA0FRITS.IPIDX).

17.2.12.7 FLXA0FRUIR — FlexRay User Input Transfer Request Register

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: ERAY + 0818_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	UIDX[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 17.81 FLXA0FRUIR Register Contents

Bit Position	Bit Name	Function
31 to 8	—	Reserved These bits are always read as 0. When writing, always write 0.
7 to 0	UIDX[7:0]	User Requested Input Index Input pointer table index requested for input transfer

(1) FLXA0FRUIR.UIDX

User Requested Input Index

The user can only write to these bits when FLXA0FRITS.UIRP is ‘0’.

The user should not write to this register when FLXA0FRITS.ITS is ‘0’.

The user should not write to this register when FLXA0FRITS.UIRP is ‘1’.

The user should not write to this register when FLXA0FRITS.IQH is ‘1’.

The user should only write FLXA0FRITC.ITM + 1 to this register.

This value configures the user input pointer table index requested for input transfer.

The address to the input data structure has to be provided in the input pointer table at the index UIDX before writing to this register.

When writing to this register, the requested input data structure will be transferred from input data structure position to the FlexRay internal message RAM.

In opposite to queued input transfers, the related DA flag in the FLXA0FRDA register is not affected by the user input transfer.

17.2.12.8 FLXA0FRUOR — FlexRay User Output Transfer Request Register

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: ERAY + $081C_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	URDS	—	—	UMBNR[6:0]						
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 17.82 FLXA0FRUOR Register Contents

Bit Position	Bit Name	Function
31 to 10	—	Reserved These bits are always read as 0. When writing, always write 0.
9	URDS	User Request Read Data Section 0: Data section is not transferred 1: Data section is transferred
8, 7	—	Reserved These bits are always read as 0. When writing, always write 0.
6 to 0	UMBNR[6:0]	User Requested Output Message Buffer Number Message buffer number requested for output transfer

(1) FLXA0FRUOR.URDS

User Request Read Data Section

The user can only write to this bit when FLXA0FROTS.UORP is ‘0’.

The user should not write to this register when FLXA0FROTS.OTS is ‘0’.

The user should not write to this register when FLXA0FROTS.UORP is ‘1’.

0: Data section is not transferred
The data section of the message buffer selected by the bits UMBNR is not requested

1: Data section is transferred
The data section of the message buffer selected by the bits UMBNR is requested

(2) FLXA0FRUOR.UMBNR

User Requested output Message Buffer Number

The user can only write to these bits when FLXA0FROTS.UORP is ‘0’.

The user should not write to this register when FLXA0FROTS.OTS is ‘0’.

The user should not write to this register when FLXA0FROTS.UORP is ‘1’.

The user should restrict this bit to dedicated receive and transmit buffers when the FlexRay module is not in the CONFIG state.

When writing to this register, the header sections and the optional data section (configurable by URDS) of the requested message buffer will be transferred from the FlexRay internal message RAM to the output data structure position defined by the output structure data pointer in the output pointer table.

17.2.12.9 FLXA0FRAHBC — FlexRay H-Bus Configuration Register

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: ERAY + 0840_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	HPROT[3:0]			
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Table 17.83 FLXA0FRAHBC Register Contents

Bit Position	Bit Name	Function
31 to 4	—	Reserved These bits are always read as 0. When writing, always write 0.
3 to 0	HPROT[3:0]	Protection Control

(1) FLXA0FRAHBC.HPROT

Protection Control

The user should only write to this register when FLXA0FRITS.ITS is ‘0’ and when FLXA0FROTS.OTS is ‘0’.

This register configures the value assigned to the H-Bus protection control signal.

17.2.13 Data Transfer Status Register

17.2.13.1 FLXA0FRITS — FlexRay Input Transfer Status Register

Do not rewrite this register using bit-manipulation instruction.

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: ERAY + 0820_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	IGIDX[6:0]							—	IPIDX[6:0]						
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	IQFP	—	—	IQEIS	IQFIS	—	—	—	—	—	UIRP	IQH	ITS
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R	R

Table 17.84 FLXA0FRITS Register Contents

Bit Position	Bit Name	Function
31	—	Reserved This bit is always read as 0. When writing, always write 0.
30 to 24	IGIDX[6:0]	Input Queue Get Index Represents the get index of the input pointer table
23	—	Reserved This bit is always read as 0. When writing, always write 0.
22 to 16	IPIDX[6:0]	Input Queue Put Index Represents the put index of the input pointer table
15 to 13	—	Reserved These bits are always read as 0. When writing, always write 0.
12	IQFP	Input Queue Full Condition Pending 0: Entries in the input queue are available 1: All entries in the input queue are occupied
11, 10	—	Reserved These bits are always read as 0. When writing, always write 0.
9	IQEIS	Input Queue Empty Interrupt Status 0: No input queue empty condition detected 1: Input queue empty condition detected
8	IQFIS	Input Queue Full Interrupt Status 0: No input queue full condition detected 1: Input queue full condition detected
7 to 3	—	Reserved These bits are always read as 0. When writing, always write 0.
2	UIRP	User Input Transfer Request Pending 0: No user input transfer request pending 1: User input transfer request pending
1	IQH	Input Queue Halted 0: Input queue not halted 1: Input queue halted
0	ITS	Input Transfer Status 0: Disabled 1: Enabled

(1) FLXA0FRITS.IGIDX

Input Queue Get Index

These bits are only valid when FLXA0FRITS.IQH is ‘1’

These bits represent the input pointer index the input queue handler will transfer next.

Valid values are 00_H to FLXA0FRITC.ITM.

The get index is incremented when the input data structure has been transferred from the local RAM/ global RAM and the related DA flag in the FLXA0FRDA register is cleared.

The index is set to 00_H when FLXA0FRITS.ITS changes from ‘0’ to ‘1’.

(2) FLXA0FRITS.IPIDX

Input Queue Put Index

These bits represent the index where the next input data structure pointer in the input pointer table should be stored.

Valid values are 00_H to FLXA0FRITC.ITM.

After reaching the maximum value the put index continues from 00_H.

The index is incremented when writing to FLXA0FRIQC.IMBNR.

The index is set to 00_H when FLXA0FRITS.ITS changes from ‘0’ to ‘1’.

(3) FLXA0FRITS.IQFP

Input Queue Full Condition Pending

This bit represents that the input queue is full.

There should be no further input transfer requests, by writing to FLXA0FRIQC.IMBNR, as long as FLXA0FRITS.IQFP is ‘1’.

[Clearing condition]

This bit is cleared when there is one free entry in the input queue.

[Setting condition]

This bit is set when all entries in the input queue are occupied.

(4) FLXA0FRITS.IQEIS

Input Queue Empty Interrupt Status

Writing ‘0’ has no effect on the bit value.

If enabled in FLXA0FRITC.IQEIE the input queue empty interrupt is generated when FLXA0FRITS.IQEIS is ‘1’.

[Clearing condition]

This bit is cleared when writing a ‘1’ to FLXA0FRITS.IQEIS.

This bit is cleared when FLXA0FRITS.ITS changes from ‘0’ to ‘1’.

[Setting condition]

This bit is set when all pending input transfers have been processed and consequently the input queue becomes empty.

(5) FLXA0FRITS.IQFIS

Input Queue Full Interrupt Status

Writing ‘0’ has no effect on the bit value.

If enabled in FLXA0FRITC.IQFIE the input queue full interrupt is generated when FLXA0FRITS.IQFIS is ‘1’.

This flag is intended as interrupt status flag. It does not represent the current input queue status; for this status refer to FLXA0FRITS.IQFP.

[Clearing condition]

This bit is cleared when writing a ‘1’ to FLXA0FRITS.IQFIS.

This bit is cleared when FLXA0FRITS.ITS changes from ‘0’ to ‘1’.

[Setting condition]

This bit is set when all entries in the input queue are occupied.

(6) FLXA0FRITS.UIRP

User Input Transfer Request Pending

This bit represents that a user input transfer is still pending.

There should be no further write access to FLXA0FRUIR.UIDX when this bit is ‘1’.

[Clearing condition]

This bit is cleared when the user input transfer request is processed by the input transfer handler.

[Setting condition]

This bit is set when writing to FLXA0FRUIR.UIDX.

(7) FLXA0FRITS.IQH

Input Queue Halted

This bit represents the status of the input queue.

There should be no further write access to FLXA0FRUIR.UIDX when this bit is ‘1’.

[Clearing condition]

This bit is cleared when FLXA0FRITC.IQHR is set to ‘0’.

[Setting condition]

This bit is set immediately when the FLXA0FRITC.IQHR is set to ‘1’ and there is no ongoing input transfer.

This bit is set only after an ongoing input transfer has been completed and FLXA0FRITC.IQHR is set to ‘1’.

(8) FLXA0FRITS.ITS

Input Transfer Status

This bit represents the status of the input queue handler.

While this bit is '1', there should be no read or write access to the address area ERAY + 0400_H to ERAY + $05FF_H$ and there should be no CLEAR_RAMS command applied to FLXA0FRSUCC1.CMD.

The input transfer queue indexes and related status flags are set to '0' when FLXA0FRITS.ITS changes from '0' to '1'.

[Clearing condition]

This bit is cleared immediately when FLXA0FRITC.ITE is set to '0' and there are no pending input transfers

This bit is cleared after all pending requests have been processed and FLXA0FRITC.ITE is '0'.

[Setting condition]

This bit is set when FLXA0FRITC.ITE is set to '1'.

17.2.13.2 FLXA0FROTS — FlexRay Output Transfer Status Register

Do not rewrite this register using bit-manipulation instruction.

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: ERAY + 0824_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	FFL[5:0]						—	—	—	FGIDX[4:0]				
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	FWP	OWP	FDA	—	FWS	OWS	FIS	OTIS	—	—	—	—	—	UORP	—	OTS
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Table 17.85 FLXA0FROTS Register Contents (1/2)

Bit Position	Bit Name	Function
31, 30	—	Reserved These bits are always read as 0. When writing, always write 0.
29 to 24	FFL[5:0]	FIFO Fill Level Represent the number of unprocessed output FIFO structures
23 to 21	—	Reserved These bits are always read as 0. When writing, always write 0.
20 to 16	FGIDX[4:0]	FIFO Get Index Represent the get index in the FIFO pointer table
15	FWP	FIFO Transfer Warning Condition Pending 0: No FIFO transfer warning condition pending 1: FIFO transfer warning condition pending
14	OWP	Output Transfer Warning Condition Pending 0: No output transfer warning condition pending 1: Output transfer warning condition pending
13	FDA	FIFO Data Available 0: No available FIFO structures 1: FIFO structures available at current FLXA0FROTS.FGIDX index
12	—	Reserved This bit is always read as 0. When writing, always write 0.
11	FWIS	FIFO Transfer Warning Interrupt Status 0: No FIFO transfer warning condition detected 1: FIFO transfer warning condition detected
10	OWIS	Output Transfer Warning Interrupt Status 0: No output transfer warning condition detected 1: Output transfer warning condition detected
9	FIS	FIFO Transfer Interrupt Status 0: No FIFO structure updated in local RAM/global RAM 1: FIFO structure updated in local RAM/global RAM
8	OTIS	Output Transfer Interrupt Status 0: No output structure updated in local RAM/global RAM 1: Output structure updated in local RAM/global RAM
7 to 3	—	Reserved These bits are always read as 0. When writing, always write 0.

Table 17.85 FLXA0FROTS Register Contents (2/2)

Bit Position	Bit Name	Function
2	UORP	User Output Transfer Request Pending 0: No user output transfer request pending 1: User output transfer request pending
1	—	Reserved This bit is always read as 0. When writing, always write 0.
0	OTS	Output Transfer Status 0: Disabled 1: Enabled

(1) FLXA0FROTS.FFL

FIFO Fill Level

These bits represent the number of available output FIFO structures in the local RAM/global RAM.

Valid values are 00_H to FLXA0FROTC.FTM + 1.

The value 00_H represents that the FIFO is empty.

The value FLXA0FROTC.FTM + 1 represents that the FIFO is full and no further FIFO transfers will be done.

These bits are incremented when a FIFO data structure has been transferred from the FlexRay internal FIFO into the local RAM/global RAM.

The FIFO fill level is decremented when the user releases a FIFO data structure in the local RAM/global RAM by writing '1' to FLXA0FROTS.FDA.

The FIFO fill level is set to 00_H when the bit FLXA0FROTS.OTS changes from '0' to '1'.

(2) FLXA0FROTS.FGIDX

FIFO Get Index

These bits represent the index where the current output data structure pointer in the FIFO pointer table is available for reading.

Valid values are 00_H to FLXA0FROTC.FTM.

After reaching the maximum value the get index continues from 00_H.

The index is incremented when a FIFO data structure is released by writing '1' to FLXA0FROTS.FDA.

The index is set to 00_H when FLXA0FROTS.OTS changes from '0' to '1'.

(3) FLXA0FROTS.FWP

FIFO Transfer Warning Condition Pending

This bit represents the FIFO transfer warning condition.

[Clearing condition]

This bit is cleared when there are free output data structures (FLXA0FROTS.FFL ≤ FLXA0FROTC.FTM).

This bit is cleared when the FLXA0FROTS.OTS changes from '0' to '1'.

[Setting condition]

This bit is set when the output transfer handler detects a transfer condition for FIFO message buffers but there are no free output data structures (FLXA0FROTS.FFL = FLXA0FROTC.FTM + 1).

(4) FLXA0FROTS.OWP

Output Transfer Warning Condition Pending

This bit represents the output transfer warning condition.

[Clearing condition]

This bit is cleared, when all output structure pointers that have a pending output handler transfer condition detected, are released (for dedicated transmit and receive message buffers or a user output transfer request).

[Setting condition]

This bit is set when the output transfer handler detects a transfer condition (for dedicated transmit and receive message buffers or a user output transfer request) but the related output structure pointer was not yet released by the application (data available flag is still set to ‘1’).

This bit is set when the output transfer handler detects a transfer condition for dedicated transmit and receive message buffers but there is a pending input transfer for the same message buffer (data available flag is set to ‘1’ due to the input transfer request).

(5) FLXA0FROTS.FDA

FIFO Data Available

Writing '0' has no effect on the bit value.

When this bit is ‘1’, the next valid output data structure is available.

The related data structure pointer is in the FIFO pointer table at FLXA0FROTS.FGIDX.

Writing ‘1’ to FLXA0FROTS.FDA

- increments FLXA0FROTS.FGIDX and
- decrements the FIFO fill level (FLXA0FROTS.FFL)

If there are still unprocessed data structures, FLXA0FROTS.FDA remains ‘1’.

[Clearing condition]

This bit is cleared when writing ‘1’ to FLXA0FROTS.FDA and the FIFO fill level becomes 00_H.

This bit is cleared when the FLXA0FROTS.OTS changes from ‘0’ to ‘1’.

[Setting condition]

This bit is set when there is at least one FIFO data structure available in the local RAM/global RAM.

(6) FLXA0FROTS.FWIS

FIFO Transfer Warning Interrupt Status

Writing '0' has no effect on the bit value.

If the FLXA0FROTC.FWIE is enabled, the FIFO transfer warning interrupt is generated when FLXA0FROTS.FWIS is ‘1’.

[Clearing condition]

This bit is cleared when writing a ‘1’ to FLXA0FROTS.FWIS.

This bit is cleared when the bit FLXA0FROTS.OTS changes from ‘0’ to ‘1’.

[Setting condition]

This bit is set when the output transfer handler detects a transfer condition for FIFO message buffers but there are no free output data structures (FLXA0FROTS.FFL = FLXA0FROTC.FTM + 1).

(7) FLXA0FROTS.OWIS

Output Transfer Warning Interrupt Status

Writing ‘0’ has no effect on the bit value.

If enabled in FLXA0FROTC.OWIE the FIFO transfer warning interrupt is generated when FLXA0FROTS.OWIS is ‘1’.

[Clearing condition]

This bit is cleared when writing a ‘1’ to FLXA0FROTS.OWIS.

This bit is cleared when the bit FLXA0FROTS.OTS changes from ‘0’ to ‘1’.

[Setting condition]

This bit is set when the output transfer handler detects a transfer condition (for dedicated transmit and receive message buffers or a user output transfer request) but the related output structure pointer was not yet released by the application (data available flag is still set to ‘1’).

This bit is set when the output transfer handler detects a transfer condition for dedicated transmit and receive message buffers but there is a pending input transfer for the same message buffer (data available flag is set to ‘1’ due to the input transfer request).

(8) FLXA0FROTS.FIS

FIFO Transfer Interrupt Status

Writing ‘0’ has no effect on the bit value.

If enabled in FLXA0FROTC.FIE the FIFO transfer interrupt is generated when FLXA0FROTS.FIS is ‘1’.

[Clearing condition]

This bit is cleared when writing a ‘1’ to FLXA0FROTS.FIS.

This bit is cleared when the bit FLXA0FROTS.OTS changes from ‘0’ to ‘1’.

[Setting condition]

This bit is set when a FIFO data structure is updated by the transfer handler or the FFL bit changes from 00_H to 01_H.

(9) FLXA0FROTS.OTIS

Output Transfer Interrupt Status

Writing ‘0’ has no effect on the bit value.

If enabled in FLXA0FROTC.OIE the output transfer interrupt is generated when FLXA0FROTS.OTIS is ‘1’.

[Clearing condition]

This bit is cleared when writing a ‘1’ to FLXA0FROTS.OTIS.

This bit is cleared when the bit FLXA0FROTS.OTS changes from ‘0’ to ‘1’.

[Setting condition]

This bit is set when an output data structure is updated by the transfer handler (from a dedicated transmit or receive message buffer or by a user output transfer request).

(10) FLXA0FROTS.UORP

User Output Transfer Request Pending

This bit represents that a user output transfer is still pending.

There should be no further write access to FLXA0FRUOR.UMBNR when this bit is ‘1’.

[Clearing condition]

This bit is cleared when the user output transfer request is processed by the output transfer handler.

This bit is cleared when the bit FLXA0FROTS.OTS changes from ‘0’ to ‘1’.

[Setting condition]

This bit is set when writing to FLXA0FRUOR.UMBNR.

(11) FLXA0FROTS.OTS

Output Transfer Status

This bit represents the status of the output transfer handler.

While this bit is ‘1’, no read or write access to the address area ERAY + 0600_H to ERAY + $07FF_H$ cannot be made and no CLEAR_RAMS command to the FLXA0FRSUCC1.CMD cannot be provided.

While this bit is ‘1’, the user cannot change the E-Ray message RAM configuration by writing to the FLXA0FRMRC register.

The output hander transfer indexes and related status flags are set to ‘0’ when FLXA0FROTS.OTS changes from ‘0’ to ‘1’.

[Clearing condition]

This bit is cleared immediately when FLXA0FROTC.OTE is set to ‘0’ and there are no ongoing output transfers.

This bit is cleared after an ongoing transfer has been completed and FLXA0FROTC.OTE is ‘0’.

[Setting condition]

This bit is set when FLXA0FROTC.OTE is set to ‘1’.

17.2.13.3 FLXA0FRAES — FlexRay Access Error Status Register

Do not rewrite this register using bit-manipulation instruction.

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: ERAY + 0828_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	MAE	FAE	OAE	IAE	EIDX[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Table 17.86 FLXA0FRAES Register Contents

Bit Position	Bit Name	Function
31 to 12	—	Reserved These bits are always read as 0. When writing, always write 0.
11	MAE	Multiple Access Errors 0: No multiple access errors occurred 1: Multiple access errors occurred
10	FAE	FIFO Transfer Access Error 0: No access error occurred during FIFO transfer 1: Access error occurred during FIFO transfer
9	OAE	Output Transfer Access Error 0: No access error occurred during output transfer 1: Access error occurred during output transfer
8	IAE	Input Transfer Access Error 0: No access error occurred during input transfer 1: Access error occurred during input transfer
7 to 0	EIDX[7:0]	Error Index Data structure pointer index number

(1) FLXA0FRAES.MAE

Multiple Access Errors

Writing ‘0’ has no effect on the bit value.

This bit represents that there were multiple access errors during a data transfer.

[Clearing condition]

This bit is cleared when writing a ‘1’ to FLXA0FRAES.MAE.

[Setting condition]

This bit is set when one of the bits FLXA0FRAES.FAE, FLXA0FRAES.OAE or FLXA0FRAES.IAE are set and

- an access to an protected address occurred during a FIFO data transfer or
- an access to an protected address occurred during an output data transfer or
- an access to an protected address occurred during an input data transfer

(2) FLXA0FRAES.FAE

FIFO Transfer Access Error

Writing ‘0’ has no effect on the bit value.

This bit represent that there was an access error during a FIFO data transfer.

[Clearing condition]

This bit is cleared when writing a ‘1’ to FLXA0FRAES.FAE.

[Setting condition]

This bit is set when a local RAM/global RAM access error was detected during a FIFO transfer and the bits FLXA0FRAES.OAE, FLXA0FRAES.IAE and FLXA0FRAES.MAE are ‘0’.

(3) FLXA0FRAES.OAE

Output Transfer Access Error

Writing ‘0’ has no effect on the bit value.

This bit represent that there was an access error during an output data transfer.

[Clearing condition]

This bit is cleared when writing a ‘1’ to FLXA0FRAES.OAE.

[Setting condition]

This bit is set when a local RAM/global RAM access error was detected during an output transfer and the bits FLXA0FRAES.FAE, FLXA0FRAES.IAE and FLXA0FRAES.MAE are ‘0’.

(4) FLXA0FRAES.IAE

Input Transfer Access Error

Writing '0' has no effect on the bit value.

This bit represent that there was an access error during an input data transfer.

[Clearing condition]

This bit is cleared when writing a ‘1’ to FLXA0FRAES.IAE.

[Setting condition]

This bit is set when a local RAM/global RAM access error was detected during an output transfer and the bits FLXA0FRAES.OAE, FLXA0FRAES.FAE and FLXA0FRAES.MAE are ‘0’.

(5) FLXA0FRAES.EIDX

Error Index

This value is only valid when one of the bits FLXA0FRAES.FAE, FLXA0FRAES.OAE or FLXA0FRAES.IAE is ‘1’.

When the bit FLXA0FRAES.FAE is ‘1’, FLXA0FRAES.EIDX holds the used FIFO put index when an access error has occurred.

When the bit FLXA0FRAES.OAE is ‘1’, FLXA0FRAES.EIDX holds the input pointer table get index used when an access error occurred during an input transfer or when the user request an input transfer.

When the bit FLXA0FRAES.IAE is ‘1’, FLXA0FRAES.EIDX holds the used input pointer table get index when an access error has occurred.

These bits are updated when one of the bits FLXA0FRAES.FAE, FLXA0FRAES.OAE or FLXA0FRAES.IAE is changing from ‘0’ to ‘1’.

17.2.13.4 FLXA0FRAEA — FlexRay Access Error Address Register

Access: This register can be read in 8-, 16-, or 32-bit units.

Address: ERAY + $082C_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	AEA[31:16]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	AEA[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 17.87 FLXA0FRAEA Register Contents

Bit Position	Bit Name	Function
31 to 0	AEA[31:0]	Access Error Address Address in the local RAM/global RAM when an access error has occurred

(1) FLXA0FRAEA.AEA

Access Error Address

This value is only valid when one of the bits FLXA0FRAES.FAE, FLXA0FRAES.OAE or FLXA0FRAES.IAE is ‘1’.

These bits represent the address of the access error indicated in the FLXA0FRAES register.

These bits are updated when one of the bits FLXA0FRAES.FAE, FLXA0FRAES.OAE or FLXA0FRAES.IAE is changing from ‘0’ to ‘1’.

17.2.13.5 FLXA0FRDAi — FlexRay Message Data Available Register i (i = 0 to 3)

Do not rewrite this register using bit-manipulation instruction.

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: ERAY + 0830_H to ERAY + $083C_H$ (ERAY + 0830_H + i*4)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DAb (b = (i+1)×32-1 to i×32+16)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DAb (b = i×32+15 to i×32)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 17.88 FLXA0FRDAi Register Contents

Bit Position	Bit Name	Function
31 to 0	DAb	Data Available b 0: No data available for destination 1: Data available for destination

(1) FLXA0FRDAi.DAb (b = i*32 to ((i + 1)*32)-1)

Data Available b

The user should not write a ‘1’ to bits that are ‘0’.

To maintain the status of input transfers, the user should not clear bits related to input transfers.

This register is used for input and output transfers.

Each flag corresponds to a FlexRay message buffer.

[Clearing condition]

Input transfer:

This bit is cleared when the input data structure has been transferred from the local RAM/global RAM. The data structure and the data structure pointer can be changed when the related flag is ‘0’.

Output transfer:

This bit is cleared when writing a ‘1’ to it.

[Setting condition]

Input transfer:

This bit is set when the corresponding message buffer number has been written to FLXA0FRIQC.IMBNR.

As long as this bit is ‘1’, the input data structure and the data structure pointer corresponding to this input transfer request should not be changed.

Output transfer:

This bit is set when the output data structure corresponding to this message buffer has been updated.

As long as this bit is ‘1’, the data structure is stable; no further update of the data structure will be done by the output handler. While this bit is ‘1’, the application can change the output data structure pointer in the output pointer table for this message buffer number.

17.3 Functional Description

This chapter describes the FlexRay implementation together with the related FlexRay protocol features. More information about the FlexRay protocol itself can be found in the FlexRay protocol specification.

17.3.1 FlexRay Module Operation Control

17.3.1.1 FlexRay Module Enable

After hardware reset or after the FlexRay module has been disabled (following **Section 17.3.1.2, FlexRay Module Disable**) the FlexRay module is in the reset state (FLXA0FROS.OS is '0') and the clocks of the FlexRay core module are disabled.

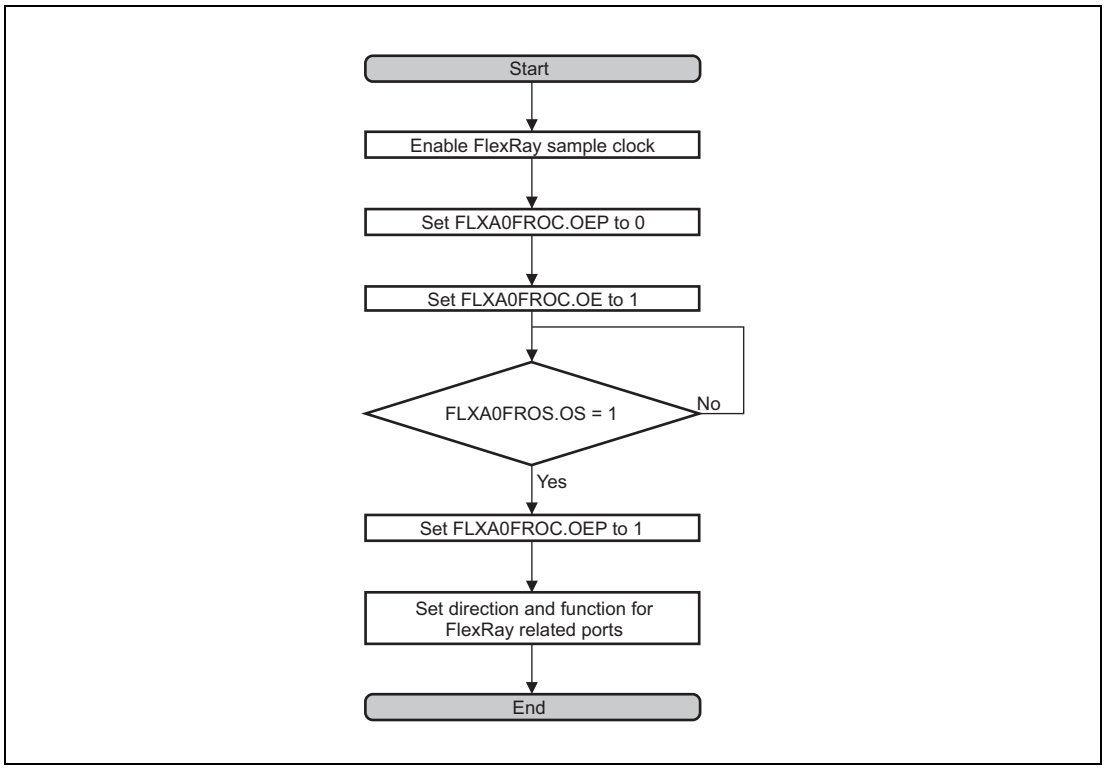


Figure 17.3 FlexRay Enable Flow

17.3.1.2 FlexRay Module Disable

The FlexRay module can be disabled at any time. However, it is recommended to disable the FlexRay module using the FLXA0FROC.OE register only when the FlexRay module is in HALT, CONFIG or DEFAULT_CONFIG state. Resetting the FlexRay module in any other state will terminate any ongoing FlexRay communication.

If the data transfer function is used, it is also required to disable this function before disabling the FlexRay module (see **Section 17.3.16.1 (1), Activation and deactivation** for suspending input transfer function and **Section 17.3.16.2 (2), Output transfer data structure** for suspending output transfer).

The following flow should be executed to disable the FlexRay module.

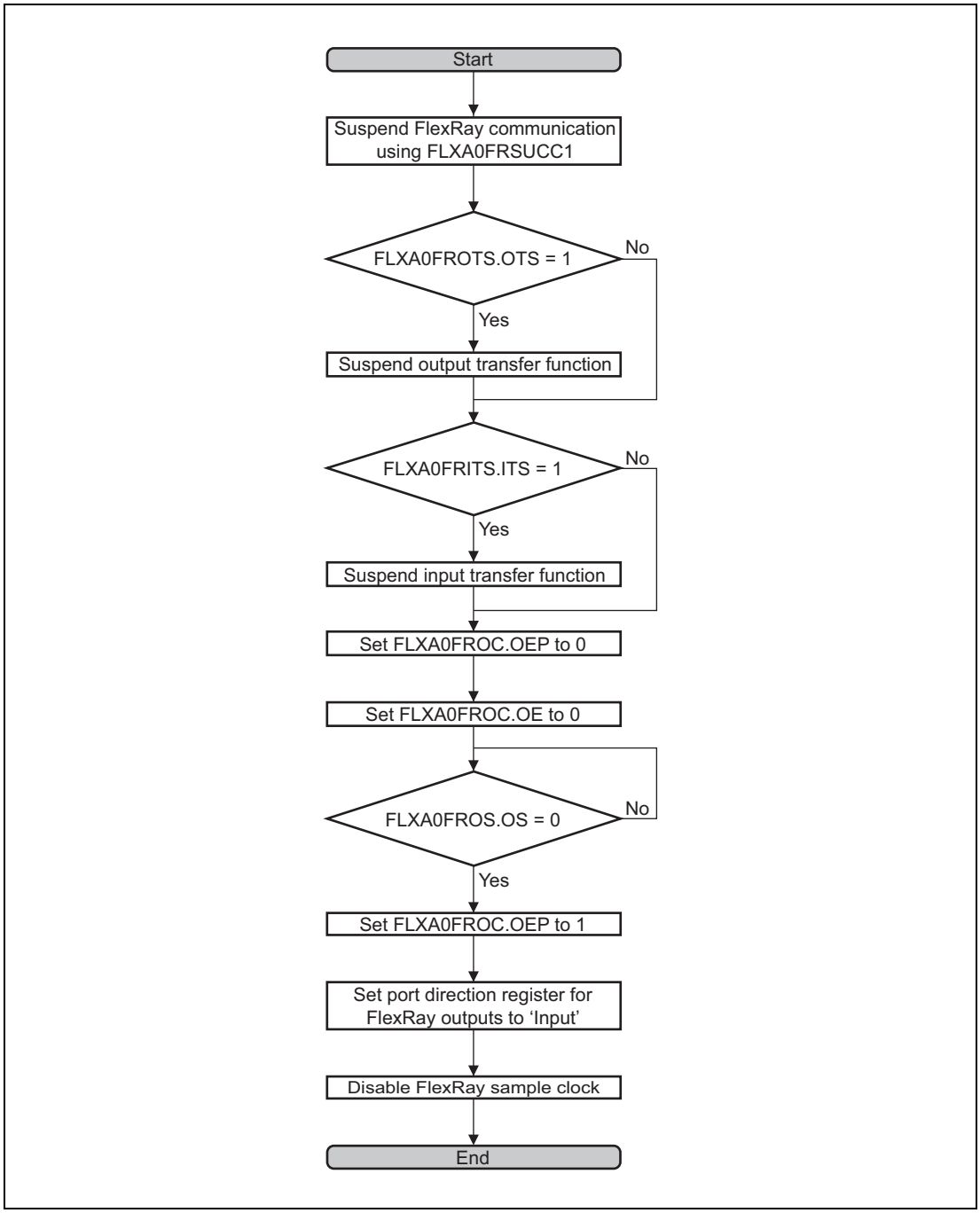


Figure 17.4 FlexRay Disable Flow

17.3.2 Communication Cycle

Communication on FlexRay networks is based on frames and symbols. The wakeup symbol (WUS) and the collision avoidance symbol (CAS) are transmitted outside the communication cycle to set up the time schedule. Frames and media access test symbols (MTS) are transmitted inside the communication cycle.

A FlexRay communication cycle consists of the following four elements:

- Static Segment
- Dynamic Segment (optional)
- Symbol Window (optional)
- Network Idle Time (NIT)

Static segment, dynamic segment, and symbol window form the Network Communication Time (NCT). For each communication channel the slot counter starts at 1 and counts up until the end of the dynamic segment is reached. Both channels share the same arbitration grid which means that they use the same synchronized macrotick.

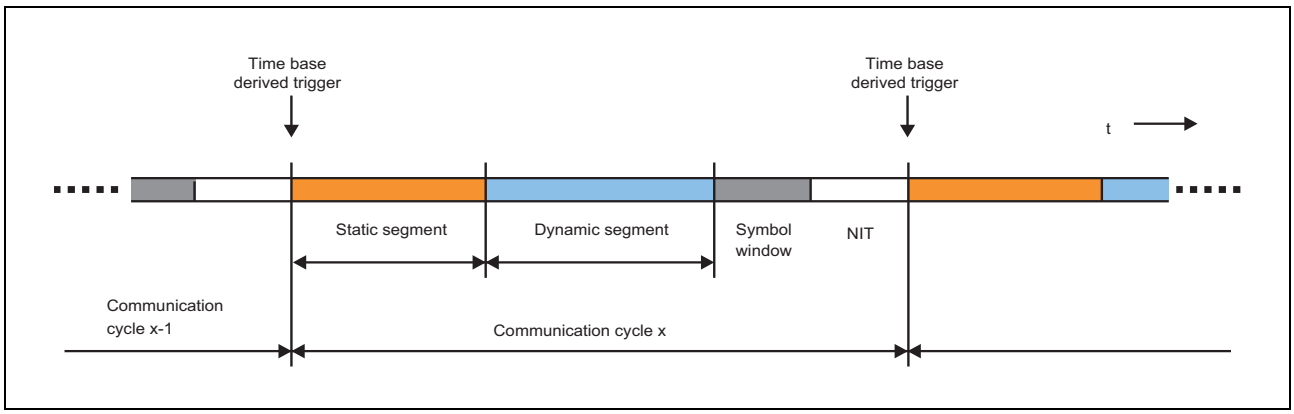


Figure 17.5 Structure of Communication Cycle

17.3.2.1 Static Segment

The Static Segment is characterized by the following features:

- Time slots of fixed length (optionally protected by bus guardian)
- Start of frame transmission at action point of the respective static slot
- Payload length same for all frames on both channels

Parameters:

Number of Static Slots (FRGTUC7.NSS)

Static Slot Length (FRGTUC7.SSL)

Payload Length Static (FRMHDC.SFDL)

Action Point Offset (FRGTUC9.APO)

17.3.2.2 Dynamic Segment

The Dynamic Segment is characterized by the following features:

- All controllers have bus access (no bus guardian protection possible)
- Variable payload length and duration of slots, different for both channels
- Start of transmission at minislot action point

Parameters:

Number of Minislots (FLXA0FRGTUC8.NMS)

Minislot Length (FLXA0FRGTUC8.MSL)

Minislot Action Point Offset (FLXA0FRGTUC9.MAPO)

Start of Latest Transmit (last minislot) (FLXA0FRMHDC.SLT)

17.3.2.3 Symbol Window

During the symbol window only one media access test symbol (MTS) may be transmitted per channel. MTS symbols are sent in NORMAL_ACTIVE state to test the bus guardian.

The symbol window is characterized by the following features:

- Send single symbol
- Transmission of the MTS symbol starts at the symbol windows action point

Parameters:

Symbol Window Action Point Offset (FLXA0FRGTUC9.APO) (same as for static slots)

Network Idle Time Start (FLXA0FRGTUC4.NIT)

17.3.2.4 Network Idle Time (NIT)

During network idle time the CC has to perform the following tasks:

- Calculate clock correction terms (offset and rate)
- Distribute offset correction over multiple macroticks after offset correction start
- Perform cluster cycle related tasks

Parameters:

Network Idle Time Start Position Setting (FLXA0FRGTUC4.NIT)

Offset Correction Start Position Setting (FLXA0FRGTUC4.OCS)

17.3.2.5 Configuration of NIT Start and Offset Correction Start

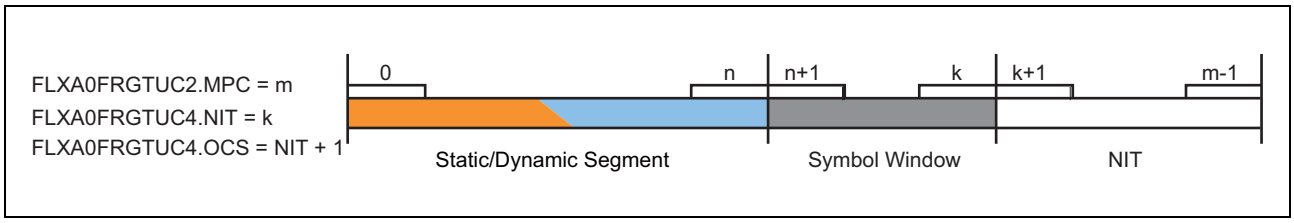


Figure 17.6 Configuration of NIT Start and Offset Correction Start

The number of macroticks per cycle gMacroPerCycle is assumed to be m. It is configured by programming FRGTUC2.MPC = m.

The static/dynamic segment starts with macrotick 0 and ends with macrotick n:

n = static segment length + dynamic segment offset + dynamic segment length - 1MT

n = gNumberOfStaticSlots o gdStaticSlot + dynamic segment offset + gNumberOfMinislots o gdMinislot - 1 MT

The static segment length is configured by FLXA0FRGTUC7.SSL and FLXA0FRGTUC7.NSS.

The dynamic segment length is configured by FLXA0FRGTUC8.MSL and FLXA0FRGTUC8.NMS.

The dynamic segment offset is:

If gdActionPointOffset ≤ gdMinislotActionPointOffset:

dynamic segment offset = 0 MT

Else if gdActionPointOffset > gdMinislotActionPointOffset:

dynamic segment offset = gdActionPointOffset -gdMinislotActionPointOffset

The NIT starts with macrotick k+1 and ends with the last macrotick of cycle m-1. It has to be configured by setting FLXA0FRGTUC4.NIT = k.

For the FlexRay module the offset correction start is required to be FLXA0FRGTUC4.OCS ≥ FLXA0FRGTUC4.NIT + 1 = k+1.

The length of symbol window results from the number of macroticks between the end of the static/ dynamic segment and the beginning of the NIT. It can be calculated by the number of macroticks (k - n).

17.3.3 Communication Modes

The FlexRay Protocol Specification defines the Time-Triggered Distributed (TT-D) mode.

17.3.3.1 Time-triggered Distributed (TT-D)

In TT-D mode the following configurations are possible:

- Pure static: Minimum 2 or more static slots + symbol window (optional)
- Mixed static/dynamic: Minimum 2 or more static slots + dynamic segment + symbol window (optional)

A minimum of two coldstart nodes needs to be configured for distributed time-triggered operation. Two fault-free coldstart nodes are necessary for the cluster startup. Each startup frame must be a sync frame, therefore all coldstart nodes are sync nodes.

17.3.4 Clock Synchronization

In TT-D mode a distributed clock synchronization is used. Each node individually synchronizes itself to the cluster by observing the timing of received sync frames from other nodes.

17.3.4.1 Global Time

Activities in a FlexRay node, including communication, are based on the concept of a global time. It is the clock synchronization mechanism that differentiates the FlexRay cluster from other node collections with independent clock mechanisms. The global time is a vector of two values; the cycle (cycle counter) and the cycle time (macrotick counter).

Cluster specific:

- Macrotick (MT) = basic unit of time measurement in a FlexRay network, a macrotick consists of an integer number of microticks (µT)
- Cycle length = duration of a communication cycle in units of macroticks (MT)

17.3.4.2 Local Time

Internally, nodes time their behavior with microtick resolution. Microticks are time units derived from the oscillator clock tick of the specific node. Therefore microticks are controller-specific units. They may have different duration in different controllers. The precision of a node's local time difference measurements is a microtick (µT).

Node specific:

- Oscillator clock → prescaler → microtick (µT)
- µT = basic unit of time measurement in a CC, clock correction is done in units of → µTs
- Cycle counter + macrotick counter = nodes local view of the global time

17.3.4.3 Synchronization Process

Clock synchronization is performed by means of sync frames. Only preconfigured nodes (sync nodes) are allowed to send sync frames. In a two-channel cluster a sync node has to send its sync frame on both channels.

For synchronization in FlexRay the following constraints have to be considered:

- Max. one sync frame per node in one communication cycle
- Max. 15 sync frames per cluster in one communication cycle
- Every node has to use a preconfigured number of sync frames (FLXA0FRGTUC2.SNM) for clock synchronization
- Minimum of two sync nodes required for clock synchronization and startup

For clock synchronization, the time difference between expected and observed arrival time of sync frames received during the static segment is measured. In a two channel cluster, the sync node has to be configured to send sync frames on both channels. The calculation of correction terms is done during NIT (offset: every cycle, rate: every odd cycle) by using an FTM algorithm. For details see FlexRay protocol specification, chapter 8.

(1) Offset (phase) Correction

- Only deviation values measured and stored in the current cycle used
- For a two channel node the smaller value will be taken
- Calculation during NIT of every communication cycle
- Offset correction value calculated in even cycles used for error checking only
- Checked against limit values
- Correction value is a signed integer number of µTs
- Correction done in odd numbered cycles, distributed over the macroticks beginning at offset correction start up to cycle end (end of NIT) to shift nodes next start of cycle (MTs lengthened or shortened)

(2) Rate (frequency) Correction

- Pairs of deviation values measured and stored in even/odd cycle pair used
- For a two channel node the average of the differences from the two channels is used
- Calculated during NIT of odd numbered cycles
- Cluster drift damping is performed using global damping value
- Checked against limit values
- Correction value is a signed integer number of µTs
- Distributed over macroticks comprising the next even/odd cycle pair (MTs lengthened or shortened)

(3) Sync Frame Transmission

Sync frame transmission is only possible from buffer 0 and 1. Message buffer 1 may be used for sync frame transmission in case that sync frames should have different payloads on the two channels. In this case bit FLXA0FRMRC.SPLM has to be programmed to ‘1’.

Message buffers used for sync frame transmission have to be configured with the key slot ID and can be (re)configured in DEFAULT_CONFIG or CONFIG state only. For nodes transmitting sync frames FLXA0FRSUCC1.TXSY must be set to ‘1’.

(4) External Clock Synchronization

During normal operation, independent clusters can drift significantly. If synchronous operation across independent clusters is desired, external synchronization is necessary; even though the nodes within each cluster are synchronized. This can be accomplished with synchronous application of host-deduced rate and offset correction terms to the clusters.

- External offset/rate correction value is a signed integer
- External offset/rate correction value is added to calculated offset/rate correction value
- Aggregated offset/rate correction term (external + internal) is not checked against configured limits

17.3.5 Error Handling

The implemented error handling concept is intended to ensure that, in case of a lower layer protocol error in one single node, communication between non-affected nodes can be maintained. In some cases, higher layer program activity is required for the CC to resume normal operation. A change of the error handling state will set FLXA0FREIR.PEMC to '1' and may trigger an interrupt to the host if enabled. The actual error mode is signaled by FLXA0FRCCEV.ERRM.

Table 17.89 Error Modes of The POC (Degradation Model)

Error Mode	Activity
ACTIVE	Full operation, State: NORMAL_ACTIVE The CC is fully synchronized and supports the cluster wide clock synchronization. The host is informed of any error condition(s) or status change by interrupt (if enabled) or by reading the error and status interrupt flags from registers FLXA0FREIR and FLXA0FRSIR.
PASSIVE	Reduced operation, State: NORMAL_PASSIVE, CC self-rescue allowed The CC stops transmitting frames and symbols, but received frames are still processed. Clock synchronization mechanisms are continued based on received frames. No active contribution to the cluster wide clock synchronization. The host is informed of any error condition(s) or status change by interrupt (if enabled) or by reading the error and status interrupt flags from registers FLXA0FREIR and FLXA0FRSIR.
COMM_HALT	Operation halted, State: HALT, CC self rescue not allowed The CC stops frame and symbol processing, clock synchronization processing, and the macrotick generation. The host has still access to error and status information by reading the error and status interrupt flags from registers FLXA0FREIR and FLXA0FRSIR. The bus drivers are disabled.

17.3.5.1 Clock Correction Failed Counter

When the Clock Correction Failed Counter reaches the "maximum without clock correction passive" limit defined by FLXA0FRSUCC3.WCP, the POC transits from NORMAL_ACTIVE to NORMAL_PASSIVE state. When it reaches the "maximum without clock correction fatal" limit defined by FLXA0FRSUCC3.WCF, it transits from NORMAL_ACTIVE or NORMAL_PASSIVE to HALT state.

The Clock Correction Failed Counter FLXA0FRCCEV.CCFC allows the host to monitor the duration of the inability of a node to compute clock correction terms after the CC passed protocol startup phase. It will be incremented by one at the end of any odd communication cycle during which either the missing offset correction FLXA0FRSFS.MOCS or the missing rate correction FLXA0FRSFS.MRCS flag is set to '1'.

The Clock Correction Failed Counter is reset to zero at the end of an odd communication cycle if neither the missing offset correction FLXA0FRSFS.MOCS nor the missing rate correction FLXA0FRSFS.MRCS flag is set to '1'.

The Clock Correction Failed Counter stops incrementing when the "maximum without clock correction fatal" value FLXA0FRSUCC3.WCF is reached (i.e. incrementing the counter at its maximum value will not cause it to wrap around back to zero). The Clock Correction Failed Counter is initialized to zero when the CC enters READY state or when NORMAL_ACTIVE state is entered.

NOTE

The transition to HALT state is prevented if FLXA0FRSUCC1.HCSE is not set to '1'.

17.3.5.2 Passive to Active Counter

The passive to active counter controls the transition of the POC from NORMAL_PASSIVE to NORMAL_ACTIVE state. FLXA0FRSUCC1.PTA defines the number of consecutive even/odd cycle pairs that must have valid clock correction terms before the CC is allowed to transit from NORMAL_PASSIVE to NORMAL_ACTIVE state. If FLXA0FRSUCC1.PTA is set to zero the CC is not allowed to transit from NORMAL_PASSIVE to NORMAL_ACTIVE state.

17.3.5.3 HALT Command

In case the host wants to stop FlexRay communication of the local node, it can bring the CC into HALT state by asserting the HALT command. This can be done by writing FLXA0FRSUCC1.CMD = “0110”. In order to shut down communication on an entire FlexRay network, a higher layer protocol is required to assure that all nodes apply the HALT command at the same time.

The POC state from which the transition to HALT state took place can be read from FLXA0FRCCSV.PSL.

When called in NORMAL_ACTIVE or NORMAL_PASSIVE state the POC transits to HALT state at the end of the current cycle. When called in any other state FLXA0FRSUCC1.CMD will be reset to “0000” = command_not_accepted and bit FLXA0FREIR.CNA is set to ‘1’. If enabled an interrupt to the host is generated.

17.3.5.4 FREEZE Command

In case the host detects a severe error condition it can bring the CC into HALT state by asserting the FREEZE command. This can be done by writing FLXA0FRSUCC1.CMD = “0111”. The FREEZE command triggers the entry of the HALT state immediately regardless of the actual POC state.

The POC state from which the transition to HALT state took place can be read from FLXA0FRCCSV.PSL.

CAUTION

When a terminal restarts transfer as a leading coldstart node after transfer was stopped by using the FREEZE or READY command, the startup frame may not be transmitted in cycle 0. This depends on the internal state of the FlexRay module. This phenomenon occurs when the startup frame is set in a slot with a number from 1 to 7. This phenomenon does not occur in a coldstart after a hardware reset.
Even if this phenomenon does occur, a second try of the coldstart will succeed. While the coldstart time will be prolonged, this phenomenon does not otherwise affect a coldstart of the FlexRay system.
To avoid this phenomenon, allocate the Startup/Sync frame to static slot 8 or a slot with a higher number.

17.3.6 Communication Controller States

17.3.6.1 Communication Controller State Diagram

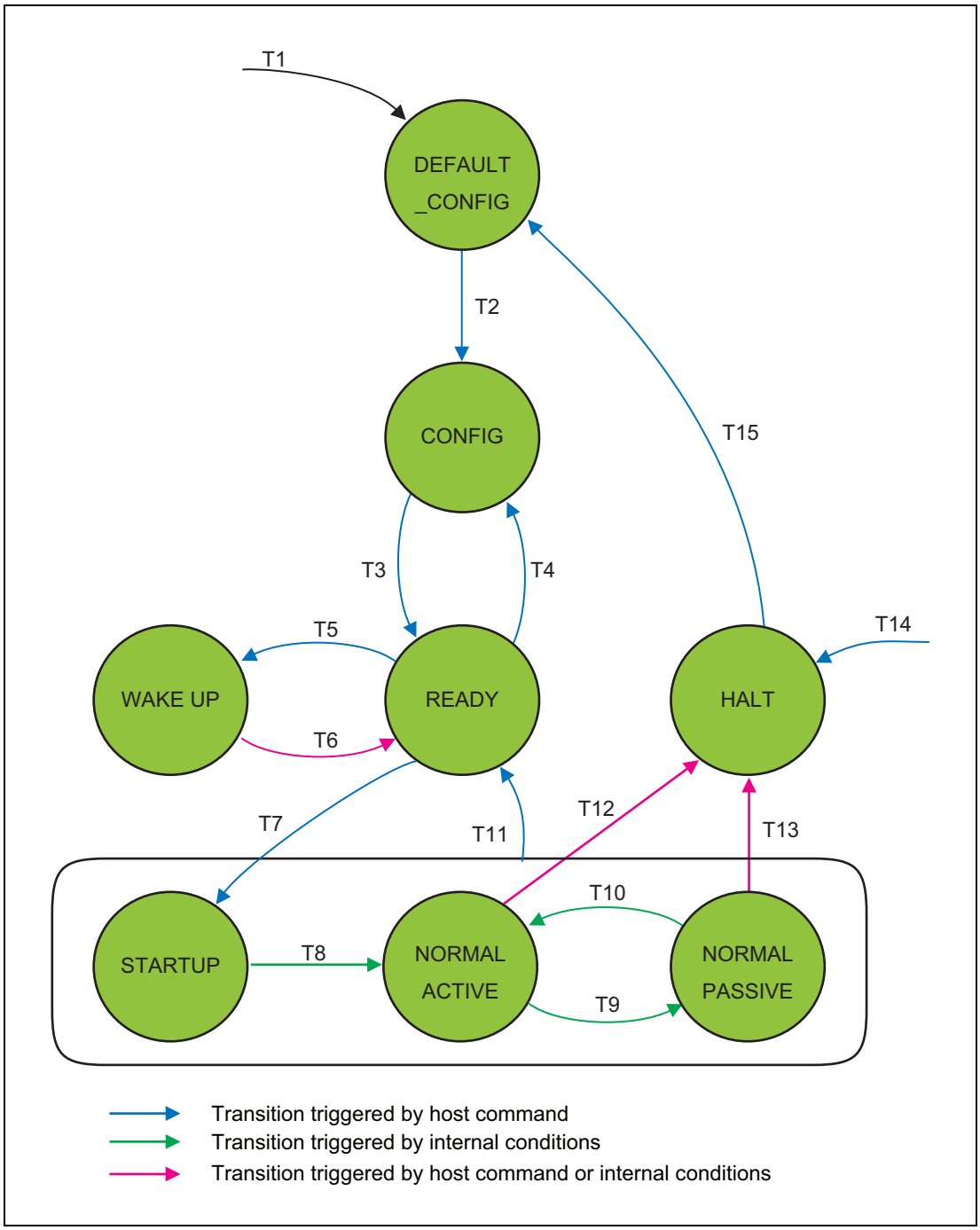


Figure 17.7 Overall state diagram of FlexRay communication controller

State transitions are controlled by reset, FLXA0FR0RXDA, FLXA0FR0RXDB, by the POC state machine, and by the CHI command vector FLXA0FRSUCC1.CMD.

The CC transits from all states to HALT state after application of the FREEZE command (FLXA0FRSUCC1.CMD = “0111”).

Table 17.90 State Transitions of FlexRay Overall State Machine

T#	Condition	From	To
1	Reset	All States	DEFAULT_CONFIG
2	Command CONFIG, FLXA0FRSUCC1.CMD = “0001”	DEFAULT_CONFIG	CONFIG
3	Unlock sequence followed by command READY, FLXA0FRSUCC1.CMD = “0010”	CONFIG	READ
4	Command CONFIG, FLXA0FRSUCC1.CMD = “0001”	READY	CONFIG
5	Command WAKEUP, FLXA0FRSUCC1.CMD = “0011”	READY	WAKEUP
6	Complete transmission of wakeup pattern or received WUP or received frame header or wakeup collision or command READY, FLXA0FRSUCC1.CMD = “0010”	WAKEUP	READY
7	Command RUN, FLXA0FRSUCC1.CMD = “0100”	READY	STARTUP
8	Successful STARTUP	STARTUP	NORMAL_ACTIVE
9	Clock Correction Failed counter reached Maximum Without Clock Correction Passive limit configured by FLXA0FRSUCC3.WCP	NORMAL_ACTIVE	NORMAL_PASSIVE
10	Number of valid correction terms reached the Passive to Active limit configured by FLXA0FRSUCC1.PTA	NORMAL_PASSIVE	NORMAL_ACTIVE
11	Command READY, FLXA0FRSUCC1.CMD = “0010”	STARTUP, NORMAL_ACTIVE, NORMAL_PASSIVE	READY
12	Clock Correction Failed counter reached Maximum Without Clock Correction Fatal limit configured by FLXA0FRSUCC3.WCF when bit FLXA0FRSUCC1.HCSE set to ‘1’ or command HALT, FLXA0FRSUCC1.CMD = “0110”	NORMAL_ACTIVE	HALT
13	Clock Correction Failed counter reached Maximum Without Clock Correction Fatal limit configured by FLXA0FRSUCC3.WCF when bit FLXA0FRSUCC1.HCSE set to ‘1’ or command HALT, FLXA0FRSUCC1.CMD = “0110”	NORMAL_PASSIVE	HALT
14	Command FREEZE, FLXA0FRSUCC1.CMD = “0111”	All States	HALT
15	Command CONFIG, FLXA0FRSUCC1.CMD = “0001”	HALT	DEFAULT_CONFIG

17.3.6.2 DEFAULT_CONFIG State

In DEFAULT_CONFIG state, the CC is stopped. All configuration registers are accessible and the pins to the physical layer are in their inactive state.

The CC enters this state

- When the reset is applied (HW reset or SW reset)
- When exiting from HALT state

To leave DEFAULT_CONFIG state the host has to write FLXA0FRSUCC1.CMD = “0001”. The CC then transits to CONFIG state.

17.3.6.3 CONFIG State

In CONFIG state, the CC is stopped. All configuration registers are accessible and the pins to the physical layer are in their inactive state. This state is used to initialize the CC configuration.

The CC enters this state

- When exiting from DEFAULT_CONFIG state
- When exiting from READY state

When the state has been entered via HALT and DEFAULT_CONFIG state, the host can analyze status information and configuration. Before leaving CONFIG state, the host has to assure that the configuration is fault-free.

To leave CONFIG state, the host has to perform the unlock sequence as described is **Section 17.2.3.1, FLXA0FRLCK — FlexRay Lock Register**. Directly after unlocking the CONFIG state the host has to write FLXA0FRSUCC1.CMD to enter the next state.

NOTE

Status bits FLXA0FRMHDS[14:0], registers FLXA0FRTXRQ1/2/3/4, and status data stored in the message RAM are not affected by the transition of the POC from CONFIG to READY state.

When the CC is in CONFIG state it is also possible to bring the CC into a power saving mode by halting the module clocks (bus clock and sample clock). To do this the host has to assure that all message RAM transfers have finished before turning off the clocks.

17.3.6.4 READY State

After unlocking CONFIG state and writing FLXA0FRSUCC1.CMD = “0010” the CC enters READY state. From this state the CC can transit to WAKEUP state and perform a cluster wake-up or to STARTUP state to perform a coldstart or to integrate into a running cluster.

The CC enters this state

- When exiting from CONFIG, WAKEUP, STARTUP, NORMAL_ACTIVE, or NORMAL_PASSIVE state by writing FLXA0FRSUCC1.CMD = “0010” (READY command).

The CC exits from this state

- To CONFIG state by writing FLXA0FRSUCC1.CMD = “0001” (CONFIG command)
- To WAKEUP state by writing FLXA0FRSUCC1.CMD = “0011” (WAKEUP command)
- To STARTUP state by writing FLXA0FRSUCC1.CMD = “0100” (RUN command)

Internal counters and the CC status flags are reset when the CC enters STARTUP state.

NOTE

Status bits FLXA0FRMHDS[14:0], registers FLXA0FRTXRQ1/2/3/4, and status data stored in the message RAM are not affected by the transition of the POC from READY to STARTUP state.

17.3.6.5 WAKEUP State

The description below is intended to help configuring wake-up for the FlexRay IP-module. A detailed description of the wake-up procedure together with the respective SDL diagrams can be found in the FlexRay protocol specification, section 7.1.

The CC enters this state

- When exiting from READY state by writing FLXA0FRSUCC1.CMD = “0011” (WAKEUP command).

The CC exits from this state to READY state

- After complete non-aborted transmission of wakeup pattern
- After WUP reception
- After detecting a WUP collision
- After reception of a frame header
- By writing FLXA0FRSUCC1.CMD = “0010” (READY command)

The cluster wakeup must precede the communication startup in order to ensure that all nodes in a cluster are awake. The minimum requirement for a cluster wakeup is that all bus drivers are supplied with power. A bus driver has the ability to wake up the other components of its node when it receives a wakeup pattern on its channel. At least one node in the cluster needs an external wakeup source.

The host completely controls the wakeup procedure. It is informed about the state of the cluster by the bus driver and the CC and configures bus guardian (if available) and CC to perform the cluster wakeup. The CC provides to the host the ability to transmit a special wakeup pattern on each of its available channels separately. The CC needs to recognize the wakeup pattern only during WAKEUP state.

Wakeup can be performed on only one channel at a time. The host has to configure the wakeup channel while the CC is in CONFIG state by writing FLXA0FRSUCC1.WUCS. The CC ensures that ongoing communication on this channel is not disturbed. The CC cannot guarantee that all nodes connected to the configured channel awake upon the transmission of the wakeup pattern, since these nodes cannot give feedback until the startup phase. The wakeup procedure enables single-channel devices in a two-channel system to trigger the wakeup, by only transmitting the wakeup pattern on the single channel to which they are connected. Any coldstart node that deems a system startup necessary will then wake the remaining channel before initiating communication startup.

The wakeup procedure tolerates any number of nodes simultaneously trying to wakeup a single channel and resolves this situation such that only one node transmits the pattern. Additionally the wakeup pattern is collision resilient, so even in the presence of a fault causing two nodes to simultaneously transmit a wakeup pattern, the resulting collided signal can still wake the other nodes.

After wakeup, the CC returns to READY state and signals the change of the wakeup status to the host by setting flag FLXA0FRSIR.WST. The wakeup status vector can be read from FLXA0FRCCSV.WSV. If a valid wakeup pattern was received also either flag FLXA0FRSIR.WUPA or flag FLXA0FRSIR.WUPB is set to ‘1’.

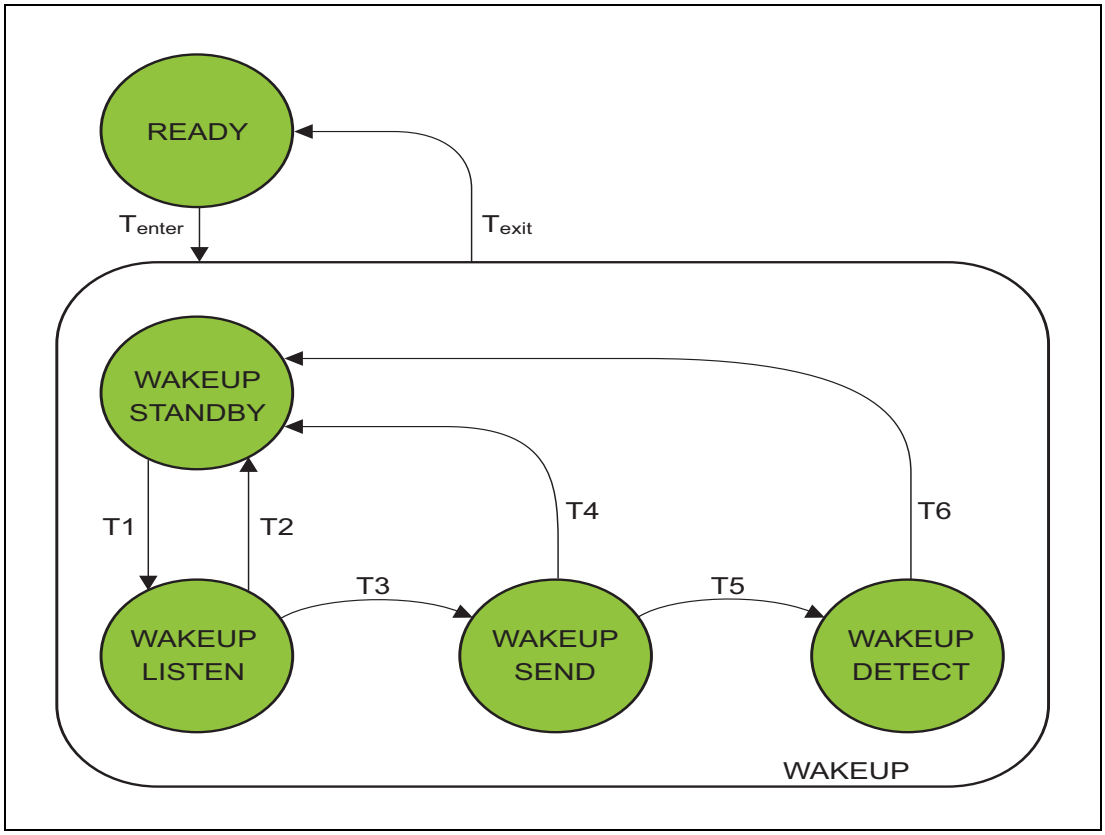


Figure 17.8 Structure of POC State WAKEUP

Table 17.91 State Transitions WAKEUP

T#	Condition	From	To
enter	Host commands change to WAKEUP state by writing FLXA0FRSUCC1.CMD = “0011” (WAKEUP command)	READY	WAKEUP
1	CHI command WAKEUP triggers wakeup FSM to transit to WAKEUP_LISTEN state	WAKEUP_STANDBY	WAKEUP_LISTEN
2	Received WUP on wakeup channel selected by bit FLXA0FRSUCC1.WUCS or frame header on either available channel	WAKEUP_LISTEN	WAKEUP_STANDBY
3	Timer event	WAKEUP_LISTEN	WAKEUP_SEND
4	Complete, non-aborted transmission of wakeup pattern	WAKEUP_SEND	WAKEUP_STANDBY
5	Collision detected	WAKEUP_SEND	WAKEUP_DETECT
6	Wakeup timer expired or WUP detected on wakeup channel selected by bit FLXA0FRSUCC1.WUCS or frame header received on either available	WAKEUP_DETECT	WAKEUP_STANDBY
exit	Wakeup completed (after T2 or T4 or T6) or host commands change to READY state by writing FLXA0FRSUCC1.CMD = “0010” (READY command). This command also resets the wakeup FSM to WAKEUP_STANDBY state	WAKEUP	READY

The WAKEUP_LISTEN state is controlled by the wakeup timer and the wakeup noise timer. The two timers are controlled by the parameters listen timeout FLXA0FRSUCC2.LT and listen timeout noise FLXA0FRSUCC2.LTN. Listen timeout enables a fast cluster wakeup in case of a noise free environment, while listen timeout noise enables wakeup under more difficult conditions regarding noise interference.

In WAKEUP_SEND state the CC transmits the wakeup pattern on the configured channel and checks for collisions. After return from wakeup the host has to bring the CC into STARTUP state by CHI command RUN.

In WAKEUP_DETECT state the CC attempts to identify the reason for the wakeup collision detected in WAKEUP_SEND state. The monitoring is limited by the expiration of listen timeout as configured by FLXA0FRSUCC2.LT. Either the detection of a wakeup pattern indicating a wakeup attempt by another node or the reception of a frame header indicating ongoing communication, causes the direct transition to READY state. Otherwise WAKEUP_DETECT is left after expiration of listen timeout; in this case the reason for wakeup collision is unknown.

The host has to be aware of possible failures of the wakeup and act accordingly. It is advisable to delay any potential startup attempt of the node having instigated the wakeup by the minimal time it takes another coldstart node to become awake and to be configured.

The FlexRay Protocol Specification recommends that two different CCs shall awake the two channels.

(1) Host activities

The host must coordinate the wakeup of the two channels and must decide whether or not to wake a specific channel. The sending of the wakeup pattern is initiated by the host. The wakeup pattern is detected by the remote BDs and signaled to their local hosts respectively.

Wakeup procedure controlled by host (single-channel wakeup):

- Configure the CC in CONFIG state
  - Select wakeup channel by programming bit FLXA0FRSUCC1.WUCS
- Check local BDs whether a WUP was received
- Activate BD of selected wakeup channel
- Command CC to enter READY state
- Command CC to start wakeup on the configured channel by writing FLXA0FRSUCC1.CMD = “0011”
  - CC enters WAKEUP
  - CC returns to READY state and signals status of wakeup attempt to the host
- Wait predefined time to allow the other nodes to wakeup and configure themselves
- Coldstart node:
  - In a dual channel cluster wait for WUP on the other channel
  - Reset coldstart inhibit flag FLXA0FRCCSV.CSI by writing FLXA0FRSUCC1.CMD = “1001” (ALLOW_COLDSTART command)
- Command CC to enter startup by writing FLXA0FRSUCC1.CMD = “0100” (RUN command)

Wakeup procedure triggered by BD:

- Wakeup recognized by BD
- BD triggers power-up of host (if required)
- BD signals wakeup event to host
- Host configures its local CC
- If necessary, host commands wakeup of second channel and waits predefined time to allow the other nodes to wakeup and configure themselves
- Host commands CC to enter STARTUP state by writing FLXA0FRSUCC1.CMD = “0100” (RUN command)

(2) Wakeup pattern (WUP)

The wakeup pattern (WUP) is composed of at least two wakeup symbols (WUS). Wakeup symbol and wakeup pattern are configured by registers FLXA0FRPRTC1 and FLXA0FRPRTC2.

- Single channel wakeup, wakeup symbol may not be sent on both channels at the same time
- Wakeup symbol collision resilient for at least two sending nodes (two overlapping wakeup symbols always recognizable)
- Wakeup symbol must be configured identical in all nodes of a cluster
- Wakeup symbol transmit low time configured by FLXA0FRPRTC2.TXL
- Wakeup symbol idle time used to listen for activity on the bus, configured by FLXA0FRPRTC2.TXI
- A wakeup pattern composed of at least two Tx-wakeup symbols needed for wakeup
- Number of repetitions configurable by FLXA0FRPRTC1.RWP (2 to 63 repetitions)
- Wakeup symbol receive window length configured by FLXA0FRPRTC1.RXW
- Wakeup symbol receive low time configured by FLXA0FRPRTC2.RXL
- Wakeup symbol receive idle time configured by FLXA0FRPRTC2.RXI

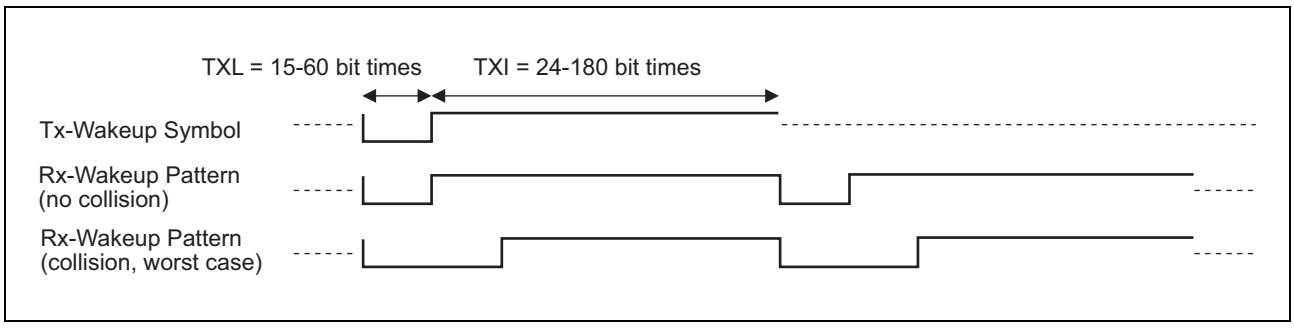


Figure 17.9 Timing of Wakeup Pattern

17.3.6.6 STARTUP State

This section describes configuration of startup for the FlexRay module. A detailed description of the startup procedure together with the respective SDL diagrams can be found in the FlexRay protocol specification, section 7.2.

Any node entering STARTUP state that has coldstart capability should assure that both channels attached have been awakened before initiating coldstart.

The required time for complete wake-up and configuration vary for each node and star. Since at least two nodes are necessary to start up the cluster communication, it is advisable to delay any potential startup attempt of the node having instigated the wakeup by the minimal amount of time it takes another coldstart node to become awake, to be configured and to enter startup. It may require several hundred milliseconds (depending on the hardware used) before all nodes and stars are completely awakened and configured.

Startup is performed on all channels synchronously. During startup, a node only transmits startup frames. Startup frames are both sync frames and null frames during startup.

A fault-tolerant, distributed startup strategy is specified for initial synchronization of all nodes. In general, a node may enter NORMAL_ACTIVE state via following paths. (see **Figure 17.10**):

- Coldstart path initiating the schedule synchronization (Leading coldstart node)
- Coldstart path joining other coldstart nodes (Following coldstart node)
- Integration path integrating into an existing communication schedule (all other nodes)

A coldstart attempt begins with the transmission of a collision avoidance symbol (CAS). Only a coldstart node that had transmitted the CAS transmits frames in the first four cycles after the CAS, it is then joined firstly by the other coldstart nodes and afterwards by all other nodes.

A coldstart node has bits FLXA0FRSUCC1.TXST and FLXA0FRSUCC1.TXSY set to ‘1’. Message buffer 0 holds the key slot ID which defines the slot number where the startup frame is send. In the frame header of the startup frame the startup frame indicator bit is set to ‘1’.

In clusters consisting of three or more nodes, at least three nodes shall be configured to be coldstart nodes. In clusters consisting of two nodes, both nodes must be coldstart nodes. At least two fault-free coldstart nodes are necessary for the cluster to startup.

Each startup frame must also be a sync frame; therefore each coldstart node will also be a sync node. The number of coldstart attempts is configured by FLXA0FRSUCC1.CSA.

A non-coldstart node requires at least two startup frames from distinct nodes for integration.

It may start integration before the coldstart nodes have finished their startup. It will not finish its startup until at least two coldstart nodes have finished their startup.

Both non-coldstart nodes and coldstart nodes start integration via the integration path as soon as they receive sync frames from which to derive the TDMA schedule information. During integration, the node has to adapt its own clock to the global clock (rate and offset) and has to make its cycle time consistent with the global schedule observable at the network. Afterwards, these settings are checked for consistency with all available network nodes. The node can only leave the integration phase and actively participate in communication when these checks are passed.

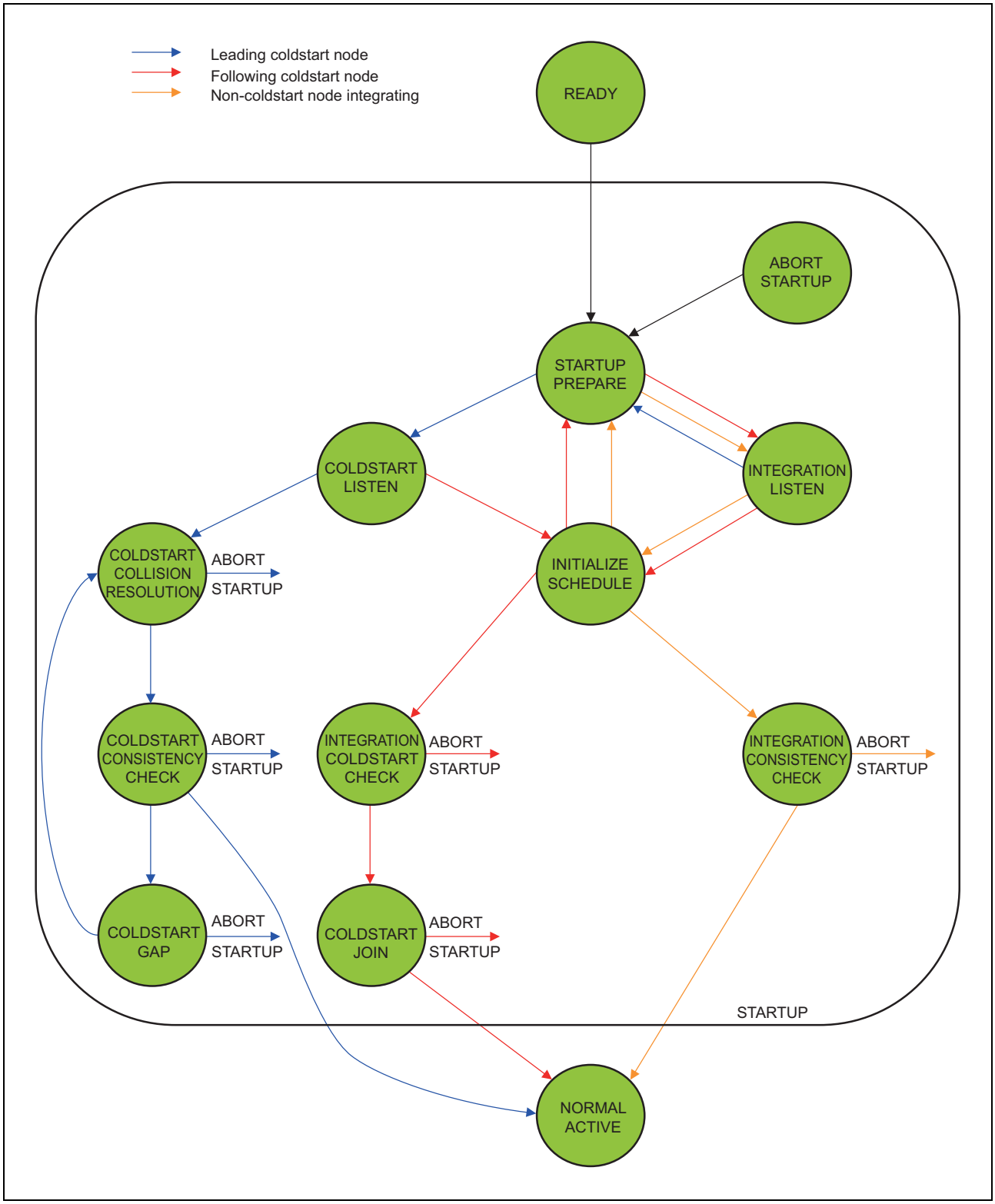


Figure 17.10 State Diagram Time-triggered Startup

(1) Coldstart Inhibit Mode

In coldstart inhibit mode the node is prevented from initializing the TDMA communication schedule. If bit FLXA0FRCCSV.CSI is set to '1', the node is not allowed to initialize the cluster communication, i.e. entering the coldstart path is prohibited. The node is allowed to integrate to a running cluster or to transmit startup frames after another coldstart node started the initialization of the cluster communication.

The coldstart inhibit bit FLXA0FRCCSV.CSI is set to '1' whenever the POC enters READY state. The bit has to be cleared by issuing the CHI command ALLOW_COLDSTART (FLXA0FRSUCC1.CMD = "1001") by a program.

(2) Startup Timeouts

The CC supplies two different µT timers supporting two timeout values, startup timeout and startup noise timeout. The two timers are started when the CC enters the COLDSTART_LISTEN state. The expiration of either of these timers causes the node to leave the initial sensing phase (COLDSTART_LISTEN state) with the intention of starting up communication.

NOTE

The startup and startup noise timers are identical with the wakeup and wakeup noise timers and use the same configuration values FLXA0FRSUCC2.LT and FLXA0FRSUCC2.LTN.

(a) Startup Timeout

The startup timeout limits the monitoring time used by a node to determine if there is already communication between other nodes or at least one coldstart node actively requesting the integration of others. The startup timer is configured by programming FLXA0FRSUCC2.LT (see **Section 17.2.6.2, FLXA0FRSUCC2 — FlexRay SUC Configuration Register 2**).

The startup timeout is:

pdListenTimeout = FLXA0FRSUCC2.LT

The startup timer is restarted upon:

- Entering the COLDSTART_LISTEN state
- Both channels reaching idle state while in COLDSTART_LISTEN state

The startup timer is stopped:

- If communication channel activity is detected on one of the configured channels while the node is in the COLDSTART_LISTEN state
- When the COLDSTART_LISTEN state is left

Once the startup timeout expires, neither an overflow nor a cyclic restart of the timer is performed. The timer status is kept for further processing by the startup state machine.

(b) Startup Noise Timeout

At the same time the startup timer is started for the first time (transition from STARTUP_PREPARE state to COLDSTART_LISTEN state), the startup noise timer is started. This additional timeout is used to improve reliability of the startup procedure in the presence of noise. The startup noise timeout is configured by programming FLXA0FRSUCC2.LTN (see **Section 17.2.6.2, FLXA0FRSUCC2 — FlexRay SUC Configuration Register 2**).

The startup noise timeout is:

pdListenTimeout * gListenNoise = FLXA0FRSUCC2.LT * (FLXA0FRSUCC2.LTN + 1)

The startup noise timer is restarted upon:

- Entering the COLDSTART_LISTEN state
- Reception of correctly decoded headers or CAS symbols while the node is in COLDSTART_LISTEN state

The startup noise timer is stopped when the COLDSTART_LISTEN state is left.

Once the startup noise timeout expires, neither an overflow nor a cyclic restart of the timer is performed. The status is kept for further processing by the startup state machine. Since the startup noise timer will not be restarted when random channel activity is detected, this timeout defines the fall-back solution that guarantees that a node will try to start up the communication cluster even in the presence of noise.

(3) Path of Leading Coldstart Node (initiating coldstart)

When a coldstart node enters COLDSTART_LISTEN, it listens to its attached channels.

If no communication is detected, the node enters the COLDSTART_COLLISION_RESOLUTION state and commences a coldstart attempt. The initial transmission of a CAS symbol is succeeded by the first regular cycle. This cycle has the number zero.

From cycle zero on, the node transmits its startup frame. Since each coldstart node may perform a coldstart attempt, it may occur that several nodes simultaneously transmit the CAS symbol and enter the coldstart path. This situation is resolved during the first four cycles after CAS transmission.

As soon as a node that initiates a coldstart attempt receives a CAS symbol or a frame header during these four cycles, it re-enters the COLDSTART_LISTEN state. Thereby, only one node remains in this path. In cycle four, other coldstart nodes begin to transmit their startup frames.

After four cycles in COLDSTART_COLLISION_RESOLUTION state, the node that initiated the coldstart enters the COLDSTART_CONSISTENCY_CHECK state. It collects all startup frames from cycle four and five and performs the clock correction. If the clock correction does not deliver any errors and it has received at least one valid startup frame pair, the node leaves COLDSTART_CONSISTENCY_CHECK and enters NORMAL_ACTIVE state.

The number of coldstart attempts that a node is allowed to perform is configured by FLXA0FRSUCC1.CSA. The number of remaining coldstarts attempts can be read from FLXA0FRCCSV.RCA. The number of remaining coldstart attempts is reduced by one for each attempted coldstart. A node may enter the COLDSTART_LISTEN state only if this value is two or larger and it may enter the COLDSTART_COLLISION_RESOLUTION state only if this value is one or larger. If the number of coldstart attempts is one, coldstart is inhibited but integration is still possible.

(4) Path of Following Coldstart Node (responding to Leading Coldstart Node)

When a coldstart node enters the COLDSTART_LISTEN state, it tries to receive a valid pair of startup frames to derive its schedule and clock correction from the leading coldstart node.

As soon as a valid startup frame has been received the INITIALIZE_SCHEDULE state is entered. If the clock synchronization can successfully receive a matching second valid startup frame and derive a schedule from this, the INTEGRATION_COLDSTART_CHECK state is entered.

In INTEGRATION_COLDSTART_CHECK state, it is assured that the clock correction can be performed correctly and that the coldstart node from which this node has initialized its schedule is still valid. The node collects all sync frames and performs clock correction in the following double-cycle. If clock correction does not signal any errors and if the node continues to receive sufficient frames from the same node it has integrated on, the COLDSTART_JOIN state is entered.

In COLDSTART_JOIN state following coldstart nodes begin to transmit their own startup frames and continue to do so in subsequent cycles. Thereby, the leading coldstart node and the nodes joining it can check if their schedules match with each other. If the clock correction signals any error, the node aborts the integration attempt. If a node in this state sees at least one valid startup frame during all even cycles in this state and at least one valid startup frame pair during all double cycles in this state, the node leaves COLDSTART_JOIN state and enters NORMAL_ACTIVE state. Thereby it leaves STARTUP at least one cycle after the node that initiated the coldstart.

(5) Path of Non-Coldstart Node

When a non-coldstart node enters the INTEGRATION_LISTEN state, it listens to its attached channels.

As soon as a valid startup frame has been received, the INITIALIZE_SCHEDULE state is entered. If the clock synchronization can successfully receive a matching second valid startup frame and derive a schedule from this, the INTEGRATION_CONSISTENCY_CHECK state is entered.

In INTEGRATION_CONSISTENCY_CHECK state the node verifies that the clock correction can be performed correctly and that enough coldstart nodes (at least 2) are sending startup frames that agree with the node's own schedule. Clock correction is activated, and if any errors are signaled, the integration attempt is aborted.

During the first even cycle in this state, either two valid startup frames or the startup frame of the node that this node has integrated on must be received; otherwise the node aborts the integration attempt.

During the first double-cycle in this state, either two valid startup frame pairs or the startup frame pair of the node that this node has integrated on must be received; otherwise the node aborts the integration attempt.

If after the first double-cycle less than two valid startup frames are received within an even cycle, or less than two valid startup frame pairs are received within a double-cycle, the startup attempt is aborted.

Nodes in this state need to detect two valid startup frame pairs for two consecutive double-cycles each to be allowed to leave STARTUP and enter NORMAL_OPERATION. Consequently, they leave startup at least one double-cycle after the node that initiated the coldstart and only at the end of a cycle with an odd cycle number.

17.3.6.7 NORMAL_ACTIVE State

As soon as the node that transmitted the first CAS symbol (resolving the potential access conflict and entering STARTUP via coldstart path) and one additional node have entered the NORMAL_ACTIVE state, the startup phase for the cluster has finished. In the NORMAL_ACTIVE state, all configured messages are scheduled for transmission. This includes all data frames as well as the sync frames. Rate and offset measurement is started in all even cycles (even/odd cycle pairs required).

In NORMAL_ACTIVE state, the CC supports regular communication functions.

- The CC performs transmissions and reception on the FlexRay bus as configured
- Clock synchronization is running
- The host interface is operational

The CC exits from that state to

- HALT state by writing FLXA0FRSUCC1.CMD = “0110” (HALT command, at the end of the current cycle)
- HALT state by writing FLXA0FRSUCC1.CMD = “0111” (FREEZE command, immediately)
- HALT state due to change of the error state from ACTIVE to COMM_HALT
- NORMAL_PASSIVE state due to change of the error state from ACTIVE to PASSIVE
- READY state by writing FLXA0FRSUCC1.CMD = “0010” (READY command)

17.3.6.8 NORMAL_PASSIVE State

NORMAL_PASSIVE state is entered from NORMAL_ACTIVE state when the error state changes from ACTIVE to PASSIVE.

In NORMAL_PASSIVE state, the node is able to receive all frames (node is fully synchronized and performs clock synchronization). Contrary to the NORMAL_ACTIVE state, the node does not actively participate in communication, i.e. neither symbols nor frames are transmitted.

In NORMAL_PASSIVE state

- The CC performs reception on the FlexRay bus
- The CC does not transmit any frames or symbols on the FlexRay bus
- Clock synchronization is running
- The host interface is operational

The CC exits from this state to

- HALT state by writing FLXA0FRSUCC1.CMD = “0110” (HALT command, at the end of the current cycle)
- HALT state by writing FLXA0FRSUCC1.CMD = “0111” (FREEZE command, immediately)
- HALT state due to change of the error state from PASSIVE to COMM_HALT
- NORMAL_ACTIVE state due to change of the error state from PASSIVE to ACTIVE. The transition takes place when FLXA0FRCCEV.PTAC equals FLXA0FRSUCC1.PTA - 1
- To READY state by writing FLXA0FRSUCC1.CMD = “0010” (READY command)

17.3.6.9 HALT State

In this state all communication (reception and transmission) is stopped.

The CC enters this state

- By writing FLXA0FRSUCC1.CMD = “0110” (HALT command) while the CC is in NORMAL_ACTIVE or NORMAL_PASSIVE state
- By writing FLXA0FRSUCC1.CMD = “0111” (FREEZE command) from all states
- When exiting from NORMAL_ACTIVE state because the FLXA0FRSUCC1.HCSE is set to ‘1’ and the clock correction failed counter reached the “maximum without clock correction fatal” limit.
- When exiting from NORMAL_PASSIVE state because the FLXA0FRSUCC1.HCSE is set to ‘1’ and the clock correction failed counter reached the “maximum without clock correction fatal” limit.

The CC exits from this state to DEFAULT_CONFIG state

- By writing FLXA0FRSUCC1.CMD = “0001” (CONFIG command)

When the CC enters HALT state, all configuration and status data is maintained for analyzing purposes.

When the host writes FLXA0FRSUCC1.CMD = “0110” (HALT command), the CC sets bit FLXA0FRCCSV.HRQ to ‘1’ and enters HALT state at the next end of cycle.

When the host writes FLXA0FRSUCC1.CMD = “0111” (FREEZE command), the CC enters HALT state immediately and sets bit FLXA0FRCCSV.FSI to ‘1’.

The POC state from which the transition to HALT state took place can be read from FLXA0FRCCSV.PSL.

17.3.7 Network Management

The accrued Network Management (NM) vector can be read from registers FLXA0FRNMV1…3. The CC performs a bit-wise OR operation over all NM vectors out of all received valid NM frames with the Payload Preamble Indicator (PPI) bit set. Only static frames may be configured to hold NM information. The CC updates the NM vector at the end of each cycle. The length of the NM vector can be configured from 0 to 12 bytes by FLXA0FRNEMC.NML. The NM vector length must be configured identically in all nodes of a cluster.

To configure a transmit buffer to send FlexRay frames with the PPI bit set as '1', bit PPIT in the header section of the respective transmit buffer has to be set to '1' via FLXA0FRWRHS1.PPIT. In addition the host has to write the NM information to the data section of the respective transmit buffer.

The evaluation of the NM vector has to be done by the application running on the host.

Section 17.3.17, Byte Alignment, for byte alignment of the received NM vector in registers FLXA0FRNMV1 to FLXA0FRNMV3.

NOTES

1. In case a message buffer is configured for transmission/reception of network management frames, the payload length configured in header 2 of that message buffer should be equal or greater than the length of the NM vector configured by FLXA0FRNEMC.NML.
2. When the CC transits to HALT state, the cycle count is not incremented and therefore the NM vector is not updated. In this case FLXA0FRNMV1…3 holds the value from the cycle before.

17.3.8 Filtering and Masking

Filtering is done by comparison of the configuration of assigned message buffers against actual slot and cycle counter values and channel ID (channel A, B). A message buffer is only updated/transmitted if the required matches occur.

Filtering is done on:

- Slot Counter
- Cycle Counter
- Channel ID

The following filter combinations for acceptance/transmit filtering are allowed:

- Slot Counter + Channel ID
- Slot Counter + Cycle Counter + Channel ID

All configured filters must match in order to store a received message in a message buffer.

NOTE

For the FIFO the acceptance filter is configured by the FIFO Rejection Filter (FLXA0FRFRF) and the FIFO Rejection Filter Mask (FLXA0FRFRFM).

A message will be transmitted in the time slot corresponding to the configured frame ID on the configured channel(s). If cycle counter filtering is enabled, the configured cycle filter value must also match.

17.3.8.1 Slot Counter Filtering

Every transmit and receive buffer contains a frame ID stored in the header section. This frame ID is compared against the actual slot counter value in order to assign receive and transmit buffers to the corresponding slot.

If two or more message buffers are configured with the same frame ID and channel ID, and if they have a matching cycle counter filter value for the same slot, then the message buffer with the lowest message buffer number is used.

17.3.8.2 Cycle Counter Filtering

Cycle counter filtering is based on the concept of a cycle set. For filtering purposes, a match is detected if any one of the elements of the cycle set is matched. The cycle set is defined by the cycle code field in header section 1 of each message buffer.

If message buffer 0 or 1 is configured to hold the startup/sync frame or the single slot frame by bits FLXA0FRSUCC1.TXST, FLXA0FRSUCC1.TXSY, and FLXA0FRSUCC1.TSM, cycle counter filtering for message buffer 0 or 1 shall be disabled.

NOTE

Sharing of a static time slot via cycle counter filtering between different nodes of a FlexRay network is not allowed.

The set of cycle numbers belonging to a cycle set is determined as described in **Table 17.92**.

Table 17.92 Definition of Cycle Set

Cycle Code	Matching Cycle Counter Values
0b000000x	All Cycles
0b000001c	Every second Cycle at (Cycle Count) mod2 = c
0b00001cc	Every fourth Cycle at (Cycle Count) mod4 = cc
0b0001ccc	Every eighth Cycle at (Cycle Count) mod8 = ccc
0b001cccc	Every sixteenth Cycle at (Cycle Count) mod16 = cccc
0b01ccccc	Every thirty-second Cycle at (Cycle Count) mod32 = ccccc
0b1cccccc	Every sixty-fourth Cycle at (Cycle Count) mod64 = cccccc

Table 17.93 below gives some examples for valid cycle sets to be used for cycle counter filtering.

Table 17.93 Examples for Valid Cycle Sets

Cycle Code	Matching Cycle Counter Values
0b0000011	1-3-5-7-…. -63
0b0000100	0-4-8-12-…. -60
0b0001110	6-14-22-30-…. -62
0b0011000	8-24-40-56
0b0100011	3-35
0b1001001	9

The received message is stored only if the cycle counter value of the cycle during which the message is received matches an element of the receive buffer's cycle set. Channel ID and frame ID must also be matched.

The frame is transmitted when the cycle code set in the transmit buffer matches the current cycle counter value. Channel ID and frame ID must also be matched.

17.3.8.3 Channel ID Filtering

There is a 2-bit channel filtering control field (CH) located in the header section of each message buffer in the message RAM. It serves as a filter for receive buffers, and as a control field for transmit buffers (see **Table 17.94**).

Table 17.94 Channel Filtering Configuration

CH[1:0]	**Transmit Buffer** transmit frame	**Receive Buffer store valid receive frame**
11	on both channels (static segment only)	received on channel A or B (store first semantically valid frame, static segment only)
10	on channel B	received on channel B
01	on channel A	received on channel A
00	Transmission prohibited	ignore frame

The contents of a transmit buffer is transmitted on the channels specified in the channel filtering field when the slot counter filtering and cycle counter filtering criteria are also met. Only in static segment a transmit buffer may be set up for transmission on both channels (CH = “11”).

Valid received frames are stored if they are received on the channels specified in the channel filtering field when the slot counter filtering and cycle counter filtering criteria are also met. Only in static segment a receive buffer may be setup for reception on both channels (CH = “11”).

NOTE

If a message buffer is configured for the dynamic segment and both bits of the channel filtering field are set to ‘1’, no frames are transmitted resp. received frames are ignored (same function as CH = “00”).

17.3.8.4 FIFO Filtering

For FIFO filtering registers FLXA0FRFRF and FLXA0FRFRFM are used. The FIFO filter consists of channel filter FLXA0FRFRF.CH, frame ID filter FLXA0FRFRF.FID, and cycle counter filter FLXA0FRFRF.CYF. Registers FLXA0FRFRF and FLXA0FRFRFM can be configured in DEFAULT_CONFIG or CONFIG state only. The filter configuration in the header section of message buffers belonging to the FIFO is ignored.

The 7-bit cycle counter filter determines the cycle set to which frame ID and channel rejection filter are applied. In cycles not belonging to the cycle set specified by FLXA0FRFRF.CYF, all frames are rejected.

A valid received frame is stored in the FIFO if channel ID, frame ID, and cycle counter are not rejected by FLXA0FRFRF and FLXA0FRFRFM register settings, and if there is no matching dedicated receive buffer.

17.3.9 Transmit Process

17.3.9.1 Static Segment

For the static segment, if there are several messages pending for transmission, the message with the frame ID corresponding to the next sending slot is selected for transmission.

The data section of transmit buffers assigned to the static segment can be updated until the end of the preceding time slot. This means that a transfer from the input buffer has to be started by writing to the input buffer command request register (FLXA0FRIBCR) latest at this time.

17.3.9.2 Dynamic Segment

In the dynamic segment, if several messages are pending, the message with the highest priority (lowest frame ID) is selected next. In the dynamic segment different slot counter sequences on channel A and channel B are possible (concurrent sending of different frame IDs on both channels).

The data section of transmit buffers assigned to the dynamic segment can be updated until the end of the preceding slot. This means that a transfer from the input buffer has to be started by writing to the input buffer command request register (FLXA0FRIBCR) latest at this time.

The start of latest transmit configured by FLXA0FRMHDC.SLT defines the maximum minislot value allowed before inhibiting new frame transmission in the dynamic segment of the current cycle.

17.3.9.3 Transmit Buffers

FlexRay message buffers can be configured as transmit buffers by programming bit CFG in the header section of the corresponding message buffer to ‘1’ via FLXA0FRWRHS1.

There exist the following possibilities to assign a transmit buffer to the CC channels:

- Static segment: channel A or channel B,
  channel A and channel B
- Dynamic segment: channel A or channel B

Message buffer 0 and 1 are dedicated to hold the startup frame, the sync frame, or the designated single slot frame as configured by FLXA0FRSUCC1.TXST, FLXA0FRSUCC1.TXSY, and FLXA0FRSUCC1.TSM respectively. In this case, they can be reconfigured in DEFAULT_CONFIG or CONFIG state only. This ensures that any node transmits at most one startup/sync frame per communication cycle. Transmission of startup/sync frames from other message buffers is not possible.

All other message buffers configured for transmission in static or dynamic segment are reconfigurable during runtime depending on the configuration of FLXA0FRMRC.SEC (see **Section 17.3.12.1, Reconfiguration of Message Buffers**). Due to the structure of the data partition in the message RAM (to be referenced by data pointer), reconfiguration of the configured payload length and the data pointer in the header section of a message buffer may lead to erroneous configurations.

If a message buffer is reconfigured (header section updated) during runtime, it may happen that this message buffer is not send out in the corresponding communication cycle.

The CC does not have the capability to calculate the header CRC. The header CRC needs to be provided to all transmit buffers by a program. If network management is required, the host has to set the PPIT bit in the header section of the respective message buffer to ‘1’ and write the network management information to the data section of the message buffer (see **Section 17.3.7, Network Management**).

The payload length field configures the payload length in 2-byte words. If the configured payload length of a static transmit buffer is shorter than the payload length configured for the static segment by

FLXA0FRMHDC.SFDL, the CC generates padding bytes to ensure that frames have proper physical length. The padding pattern is "0000_H".

NOTE

In case of an odd payload length (PLC = 1,3,5,...), the application has to write zero to the last 16 bit of the message buffers data section to ensure that the padding pattern is "0000_H".

Each transmit buffer provides a transmission mode flag TXM that allows the host to configure the transmission mode for the transmit buffer. If this bit is set, the transmitter operates in the single-shot mode. If this bit is cleared, the transmitter operates in the continuous mode.

In single-shot mode the CC resets the respective TXR flag to '0' after transmission has completed. Now the host may update the transmit buffer.

In continuous mode, the CC does not reset the respective transmission request flag TXR to '0' after successful transmission. In this case a frame is sent out each time the filter criteria match. The TXR flag can be reset to '0' by the host by writing the respective message buffer number to the FLXA0FRIBCR register while bit FLXA0FRIBCM.STXRH is set to '0'.

If two or more transmit buffers meet the filter criteria simultaneously, the transmit buffer with the lowest message buffer number will be transmitted in the respective slot.

17.3.9.4 Frame Transmission

The following steps are required to prepare a message buffer for transmission:

- Configure the transmit buffer in the message RAM via FLXA0FRWRHS1, FLXA0FRWRHS2, and FLXA0FRWRHS3
- Write the data section of the transmit buffer via FLXA0FRWRDSn
- Transfer the configuration and message data from input buffer to the message RAM by writing the number of the target message buffer to register FLXA0FRIBCR
- If configured in register FLXA0FRIBCM, the transmission request flag TXR for the respective message buffer will be set as soon as the transfer has completed, and the message buffer is ready for transmission.
- Check whether the message buffer has been transmitted by checking the respective TXR bit (TXR = '0') in the FLXA0FRTXRQ1/2/3/4 registers (single-shot mode only).

After transmission has completed, the respective TXR flag in the FLXA0FRTXRQ1/2/3/4 register is reset to '0' (single-shot mode), and, if bit MBI in the header section of the message buffer is set to '1', flag FLXA0FRSIR.TXI is set to '1'. If enabled, an interrupt is generated.

17.3.9.5 Null Frame Transmission

If in static segment the host does not set the transmission request flag to ‘1’ before transmit time, the CC transmits a null frame with the null frame indication bit set to '0' and the payload data set to zero.

In the following cases the CC transmits a null frame:

- If the message buffer with the lowest message buffer number matching the filter criteria does not have its transmission request flag set to ‘1’(TXR = ‘0’) .
- No transmit buffer configured for the slot has a cycle counter filter that matches the current cycle. In this case, no message buffer status FLXA0FRMBS is updated.

Null frames are not transmitted in the dynamic segment.

17.3.10 Receive Process

17.3.10.1 Dedicated Receive Buffers

Some FlexRay message buffers can be configured as dedicated receive buffers by programming bit CFG in the header section of the respective message buffer to ‘0’ via FLXA0FRWRHS1.

The following possibilities exist to assign a receive buffer to the CC channels:

- Static segment: channel A or channel B,
  channel A and channel B (the CC stores the first semantically valid frame)
- Dynamic segment: channel A or channel B

The CC transfers the payload data of valid received messages from the shift register of the FlexRay channel protocol controller (channel A or B) to the receive buffer with the matching filter configuration. A receive buffer stores all frame elements except the frame CRC.

All message buffers configured for reception in static or dynamic segment are reconfigurable during runtime depending on the configuration of FLXA0FRMRC.SEC (see **Section 17.3.12.1, Reconfiguration of Message Buffers**). If a message buffer is reconfigured (header section updated) during runtime, it may happen that in the respective communication cycle a received message is lost.

If two or more receive buffers meet the filter criteria simultaneously, the receive buffer with the lowest message buffer number is updated with the received message.

17.3.10.2 Frame Reception

The following steps are required to prepare a dedicated message buffer for reception:

- Configure the receive buffer in the message RAM via FLXA0FRWRHS1, FLXA0FRWRHS2, and FLXA0FRWRHS3
- Transfer the configuration from input buffer to the message RAM by writing the number of the target message buffer to register FLXA0FRIBCR

Once these steps are performed, the message buffer functions as an active receive buffer and participates in the internal acceptance filtering process which takes place every time the CC receives a message. The first matching receive buffer is updated from the received message.

If a valid payload segment was stored in the data section of a message buffer, the respective ND flag in the FLXA0FRNDAT1/2/3/4 registers is set to ‘1’, and, if bit MBI in the header section of that message buffer is set to ‘1’, flag FLXA0FRSIR.RXI is set to ‘1’. If enabled, an interrupt is generated.

In case that bit ND was already set to ‘1’ when the message handler updates the message buffer, bit FLXA0FRMBS.MLST of the respective message buffer is set to ‘1’ and the unprocessed message data is lost.

If a frame was not received, or if a null frame or a corrupted frame was received in a slot, the data section of the message buffer configured for this slot is not updated. In this case only the respective message buffer status FLXA0FRMBS is updated.

When the message handler changed the message buffer status FLXA0FRMBS in the header section of a message buffer, the respective MBC flag in the FLXA0FRMBSC1/2/3/4 registers is set to ‘1’, and if bit MBI in the header section of that message buffer is set, flag FLXA0FRSIR.MBSI is set to ‘1’. If enabled an interrupt is generated.

If the payload length of a received frame PLR is longer than the value programmed by PLC in the header section of the respective message buffer, the data field stored in the message buffer is truncated to that length.

To read a receive buffer from the message RAM via the output buffer, proceed as described in **Section 17.3.12.2 (2), Data Transfer from Message RAM to Output Buffer**.

NOTE

The ND and MBC flags are automatically cleared by the message handler when the payload data and the header of a received message have been transferred to the output buffer, respectively.

17.3.10.3 Null Frame Reception

The payload segment of a received null frame is not copied into the matching dedicated receive buffer. If a null frame has been received, only the message buffer status FLXA0FRMBS of the matching message buffer is updated from the received null frame. All bits in header 2 and 3 of the matching message buffer remain unchanged. They are updated from received data frames only.

When the message handler changed the message buffer status FLXA0FRMBS in the header section of a message buffer, the respective MBC flag in the FLXA0FRMBSC1/2/3/4 register is set to ‘1’, and if bit MBI in the header section of that message buffer is set to ‘1’, flag FLXA0FRSIR.MBSI is set to ‘1’. If enabled, an interrupt is generated.

17.3.11 FIFO Function

17.3.11.1 Description

Some message buffers can be configured as a cyclic First-In-First-Out (FIFO) buffer. The group of message buffers belonging to the FIFO is contiguous in the register map starting with the message buffer referenced by FLXA0FRMRC.FFB and ending with the message buffer referenced by FLXA0FRMRC.LCB. Up to 127 message buffers can be assigned to the FIFO.

Every valid incoming message not matching with any dedicated receive buffer but passing the programmable FIFO filter is stored into the FIFO. In this case, frame ID, payload length, receive cycle count, and the message buffer status FLXA0FRMBS of the addressed FIFO message buffer are overwritten with frame ID, payload length, receive cycle count, and the status from the received frame. It indicates that the FIFO is not empty when the RFNE flag in the FLXA0FRSIR register is set to '1', the receive FIFO fill level FLXA0FRFSR.RFFL is equal or greater than the critical level configured by FLXA0FRSIR.RFCL bit when FLXA0FRFCL.RFCL bit is 1, and A FIFO overrun has been detected when FLXA0FREIR.RFO is 1, respectively. If enabled, interrupts are generated.

If null frames are not removed by the FIFO rejection filter, the null frames will be treated like data frames when they are stored into the FIFO.

There are two index registers associated with the FIFO. The PUT Index Register (PIDX) is an index to the next available location in the FIFO. When a new message has been received, it is written into the message buffer addressed by the PIDX register. The PIDX register is then incremented and addresses the next available message buffer. If the PIDX register is incremented past the highest numbered message buffer of the FIFO, the PIDX register is loaded with the number of the first (lowest numbered) message buffer in the FIFO chain. The GET Index Register (GIDX) is used to address the next message buffer of the FIFO to be read. The GIDX register is incremented after transfer of the contents of a message buffer belonging to the FIFO to the output buffer. The PUT Index Register and the GET Index Register are not accessible by the host.

The FIFO is completely filled when the PUT index (PIDX) reaches the value of the GET index (GIDX). When the next message is written to the FIFO before the oldest message has been read, both PUT index and GET index are incremented and the new message overwrites the oldest message in the FIFO. This will set FIFO overrun flag FLXA0FREIR.RFO to ‘1’.

When the PUT index (PIDX) differs from the GET index (GIDX), it indicates that the FIFO is not empty. In this case flag FLXA0FRSIR.RFNE is set to ‘1’. This indicates that there is at least one received message in the FIFO. The FIFO empty, FIFO not empty, and the FIFO overrun states are explained in **Figure 17.11** for a three message buffer FIFO.

The programmable FlexRay FIFO Rejection Filter (FLXA0FRFRF) defines a filter pattern for messages to be rejected. The FIFO filter consists of channel filter, frame ID filter, and cycle counter filter. If bit FLXA0FRFRF.RSS is set to ‘1’, all messages received in the static segment are rejected by the FIFO. If bit FLXA0FRFRF.RNF is set to ‘1’, received null frames are not stored in the FIFO.

The FlexRay FIFO Rejection Filter Mask (FLXA0FRFRFM) specifies which bits of the frame ID filter in the FIFO Rejection Filter register are marked 'don't care' for rejection filtering.

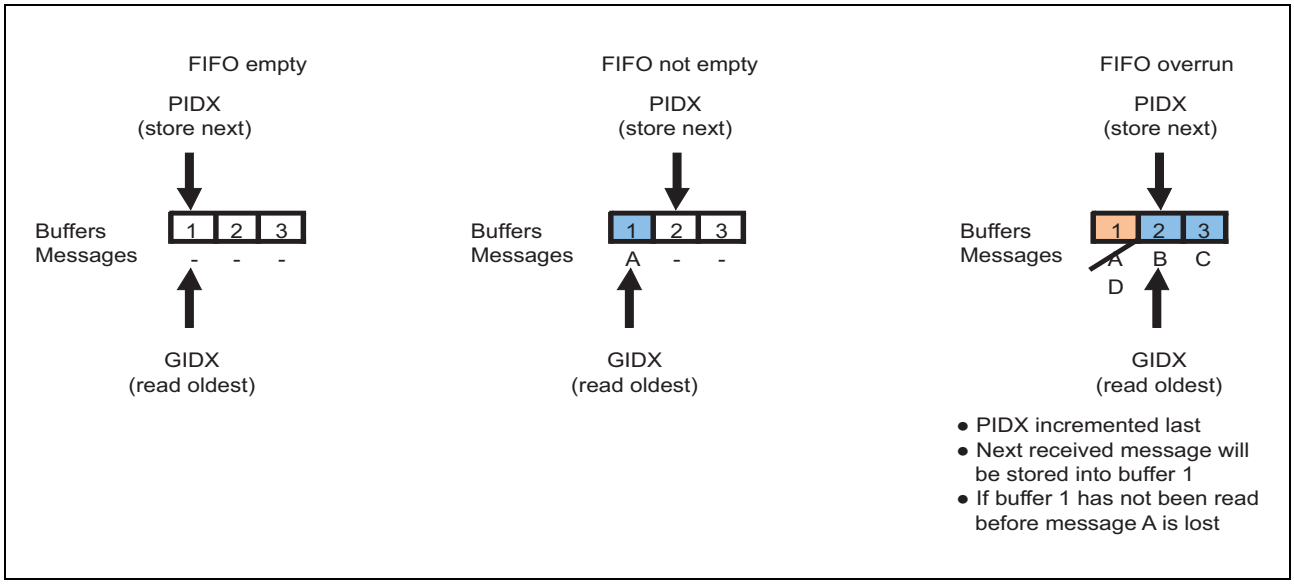


Figure 17.11 FIFO Status: Empty, Not Empty, Overrun

17.3.11.2 Configuration of the FIFO

(Re)configuration of message buffers belonging to the FIFO is only possible when the CC is in DEFAULT_CONFIG or CONFIG state. While the CC is in DEFAULT_CONFIG or CONFIG state, the FIFO function is not available.

For all message buffers belonging to the FIFO the payload length configured need to be programmed to the same value via FLXA0FRWRHS2.PLC. The data pointer to the first 32-bit word of the data section of the respective message buffer in the message RAM has to be configured via FLXA0FRWRHS3.DP.

All information required for acceptance filtering is taken from the FIFO rejection filter and the FIFO rejection filter mask registers. The values configured in the header sections of the message buffers belonging to the FIFO, with exception of DP and PLC, have no meaning.

NOTES

1. It is recommended to program the MBI bits of the message buffers belonging to the FIFO to ‘0’ via FLXA0FRWRHS1.MBI to avoid generation of RX interrupts.
2. If the payload length of a received frame is longer than the value programmed by FLXA0FRWRHS2.PLC in the header section of the respective message buffer, the data field stored in a message buffer of the FIFO is truncated to that length.

17.3.11.3 Access to the FIFO

(1) When the output buffer is used:

For FIFO access outside DEFAULT_CONFIG and CONFIG state, the host has to trigger a transfer from the message RAM to the output buffer by writing the number of the first message buffer of the FIFO (referenced by FLXA0FRMRC.FFB) to the register FLXA0FROBCR. The message handler then transfers the message buffer addressed by the GET index register (GIDX) to the output buffer. After this transfer the GET index register (GIDX) is incremented.

(2) When the data transfer function is used:

The message received in FIFO can be transferred to the local RAM/global RAM by using the output data transfer function. For the output data transfer function, see **Section 17.3.16.2, Output Data Transfer**.

17.3.12 Message Handling

The message handler controls data transfers between the input/output buffer and the message RAM and between the message RAM and the two temporary buffers.

Access to the message buffers stored in the message RAM is done under control of the message Handler state machine. This avoids conflicts between accesses of the two FlexRay channel protocol controllers and the host to the message RAM.

Frame IDs of message buffers assigned to the static segment have to be in the range from 1 to FLXA0FRGTUC7.NSS. Frame IDs of message buffers assigned to the dynamic segment have to be in the range from FLXA0FRGTUC7.NSS + 1 to 2047.

Received messages with no matching dedicated receive buffer (static or dynamic segment) are stored in the receive FIFO (if configured) if they pass the FIFO rejection filter.

Access of the host to the message buffer contents using the input or output buffer function is described in this subsection. Access to the message buffer contents using the data transfer function is mentioned in **Section 17.3.16, Usage of Data Transfer**.

17.3.12.1 Reconfiguration of Message Buffers

In case that an application needs to handle with more than 128 different messages, static and dynamic message buffers can be reconfigured during FlexRay operation. This is done by updating the header section of the respective message buffer via input buffer registers FLXA0FRWRHS1…3.

Reconfiguration has to be enabled via control bits FLXA0FRMRC.SEC in the message RAM Configuration register.

If a message buffer has not been transmitted/updated from a received frame before reconfiguration starts, the respective message is lost.

The point in time when a reconfigured message buffer is ready for transmission/reception according to the reconfigured frame ID depends on the actual state of the slot counter when the update of the header section has completed. Therefore it may happen that a reconfigured message buffer is not transmitted/updated from a received frame in the cycle where it was reconfigured.

The message RAM is scanned according to **Table 17.95** below.

Table 17.95 Scan of Message RAM

Start of Scan in Slot	Scan for Slots
1	2…15, 1 (next cycle)
8	16…23, 1 (next cycle)
16	24…31, 1 (next cycle)
24	32…39, 1 (next cycle)
…	…

A message RAM scan is terminated with the start of NIT regardless whether it has completed or not. The scan of the message RAM for slots 2 to 15 starts at the beginning of slot 1 of the actual cycle. The scan of the message RAM for slot 1 is done in the cycle before by checking in parallel to each scan of the message RAM whether there is a message buffer configured for slot 1 of the next cycle.

The number of the first dynamic message buffer is configured by FLXA0FRMRC.FDB. In case a message RAM scan starts while the CC is in dynamic segment, the scan starts with the message buffer number configured by FLXA0FRMRC.FDB.

In case a message buffer should be reconfigured to be used in slot 1 of the next cycle, the followings have to be considered:

- If the message buffer to be reconfigured for slot 1 is part of the “Static Buffers”, it will only be detected if it is reconfigured before the last message RAM scan in the static segment of the actual cycle evaluates this message buffer.
- If the message buffer to be reconfigured for slot 1 is part of the “Static + Dynamic Buffers”, it will be detected if it is reconfigured before the last message RAM scan in the actual cycle evaluates this message buffer.
- The start of NIT terminates the message RAM scan. In case the message RAM scan has not evaluated the reconfigured message buffer by this point in time, the message buffer will not be considered for the next cycle.

NOTE

Reconfiguration of message buffers may lead to the loss of messages and therefore has to be performed very carefully. In the worst case (reconfiguration in consecutive cycles), a message buffer may never be transmitted or may not be updated from a received frame.

17.3.12.2 Host Access to Message RAM

The message transfer between input buffer and message RAM as well as between message RAM and output buffer is triggered by the host by writing the number of the target and source message buffer to be accessed to FLXA0FRIBCR or FLXA0FROBCR register.

The FLXA0FRIBCM and FLXA0FROBCM registers can be used to write/read header and data section of the selected message buffer separately.

If bit FLXA0FRIBCM.STXR is set to = '1', the transmission request flag TXR of the selected message buffer is automatically set to '1' after the message buffer has been updated. If bit FLXA0FRIBCM.STXR is reset to '0', the transmission request flag TXR of the selected message buffer is reset. This can be used to stop transmission from message buffers operated in continuous mode.

Input buffer (IBF) and output buffer (OBF) are built up as a double buffer structure. One half of this double buffer structure is accessible by the host (IBF host or OBF host), while the other half (IBF shadow or OBF shadow) is accessed by the message handler for data transfers between IBF/OBF and message RAM.

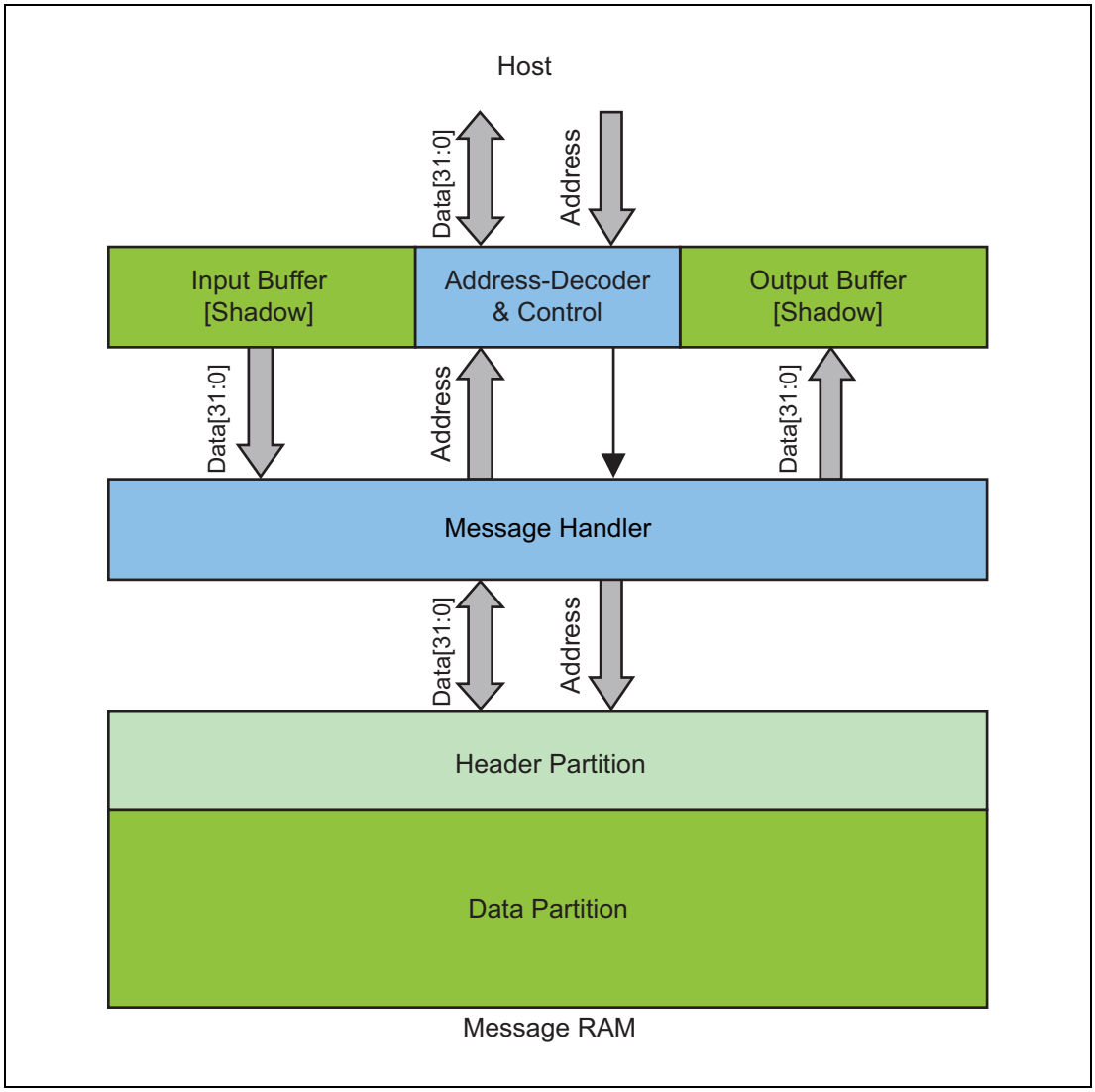


Figure 17.12 Host Access to Message RAM

(1) Data Transfer from Input Buffer to Message RAM

To configure or update a message buffer in the message RAM, the host has to write the data to FLXA0FRWRDSn and the header to FLXA0FRWRHS1…3. The specific action is selected by configuring the FlexRay input buffer command mask register FLXA0FRIBCM.

When the host writes the number of the target message buffer in the message RAM to FLXA0FRIBCR.IBRH, IBF host and IBF shadow are switched (see **Figure 17.13**).

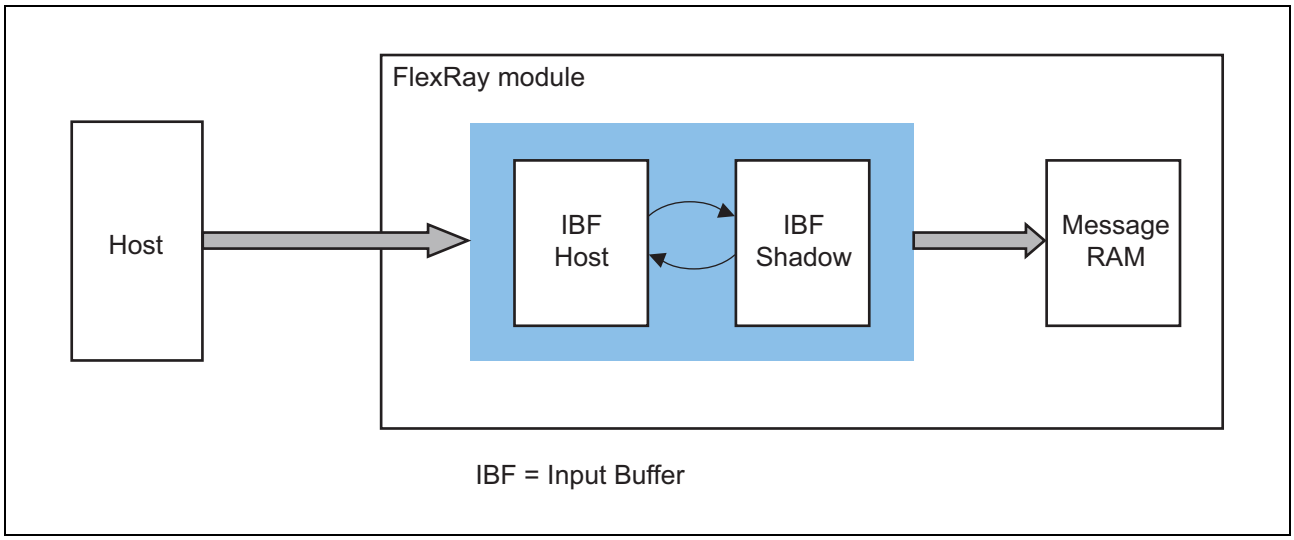


Figure 17.13 Double Buffer Structure Input Buffer

In addition the bits in the FLXA0FRIBCM and FLXA0FRIBCR registers are also switched while they hold the value in each IBF section (see **Figure 17.14**).

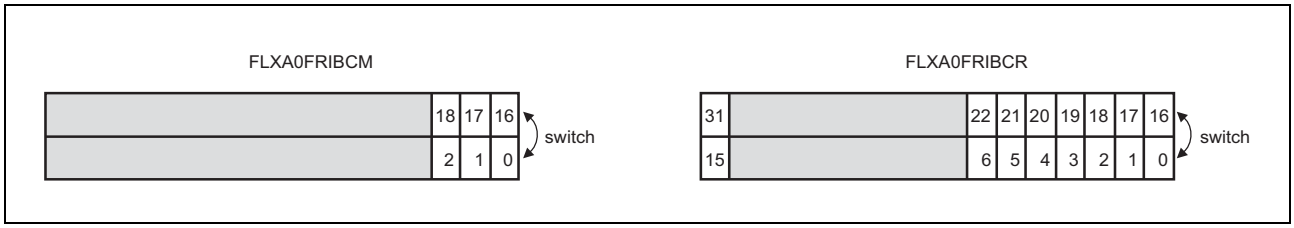


Figure 17.14 Switching of FLXA0FRIBCM and FLXA0FRIBCR Bits

With this write operation, bit FLXA0FRIBCR.IBSYS is set to ‘1’. The message handler then starts to transfer the contents of IBF shadow to the message buffer in the message RAM selected by FLXA0FRIBCR.IBRS.

While the message handler transfers the data from IBF shadow to the target message buffer in the message RAM, the host can write the next message to IBF host. After the transfer between IBF shadow and the message RAM has completed, bit FLXA0FRIBCR.IBSYS is set back to ‘0’ and the next transfer to the message RAM may be started by the host by writing the respective target message buffer number to FLXA0FRIBCR.IBRH.

If a write access to FLXA0FRIBCR.IBRH is made while FLXA0FRIBCR.IBSYS is ‘1’, FLXA0FRIBCR.IBSYH is set to ‘1’. After completion of the ongoing data transfer from IBF shadow to the message RAM, IBF host and IBF shadow are switched, FLXA0FRIBCR.IBSYH is reset to ‘0’, FLXA0FRIBCR.IBSYS remains set to ‘1’, and the next transfer to the message RAM is started. In addition the message buffer numbers stored under FLXA0FRIBCR.IBRH and FLXA0FRIBCR.IBRS and the command mask flags are also switched.

Example of an 8/16/32-bit host access sequence:

Configure or update n-th message buffer via IBF

- Wait until FLXA0FRIBCR.IBSYH is reset
- Write data section to FLXA0FRWRDSn
- Write header section to FLXA0FRWRHS1…3
- Write command mask: set FLXA0FRIBCM.STXRH, FLXA0FRIBCM.LDSH, FLXA0FRIBCM.LHSH
- Demand data transfer to target message buffer: write FLXA0FRIBCR.IBRH

Configure or update (n + 1)th message buffer via IBF

- Wait until FLXA0FRIBCR.IBSYH is reset
- Write data section to FLXA0FRWRDSn
- Write header section to FLXA0FRWRHS1…3
- Write command mask: write FLXA0FRIBCM.STXRH, FLXA0FRIBCM.LDSH, FLXA0FRIBCM.LHSH
- Demand data transfer to target message buffer: write FLXA0FRIBCR.IBRH

NOTE

Any write access to IBF while FLXA0FRIBCR.IBSYH is ‘1’ will set error flag FLXA0FREIR.IIBA to ‘1’. In this case the write access is ignored.

Table 17.96 Assignment of FLXA0FRIBCM Bits

Pos.	Access	Bit	Function
18	r	STXRS	Set Transmission Request Shadow Flag (ongoing or finished)
17	r	LDSS	Load Data Section Shadow Flag (ongoing or finished
16	r	LHSS	Load Header Section Shadow Flag (ongoing or finished)
2	r/w	STXRH	Set Transmission Request Host
1	r/w	LDSH	Load Data Section Host
0	r/w	LHSH	Load Header Section Host

Table 17.97 Assignment of FLXA0FRIBCR Bits

Pos.	Access	Bit	Function
31	r	IBSYS	IBF Busy Shadow Flag, signals ongoing transfer from IBF Shadow to message RAM
22…16	r	IBRS	IBF Request Shadow Flag, number of message buffer currently or lately updated
15	r	IBSYH	IBF Busy Host Flag, transfer request pending for message buffer referenced by IBRH
6…0	r/w	IBRH	IBF Request Host, number of message buffer to be updated next

(2) Data Transfer from Message RAM to Output Buffer

To read a message buffer from the message RAM, the host has to write to register FLXA0FROBCR to trigger the data transfer as configured in FLXA0FROBCM. After the transfer has completed, the host can read the transferred data from FLXA0FRRDDSn, FLXA0FRRDHS1…3, and FLXA0FRMBS.

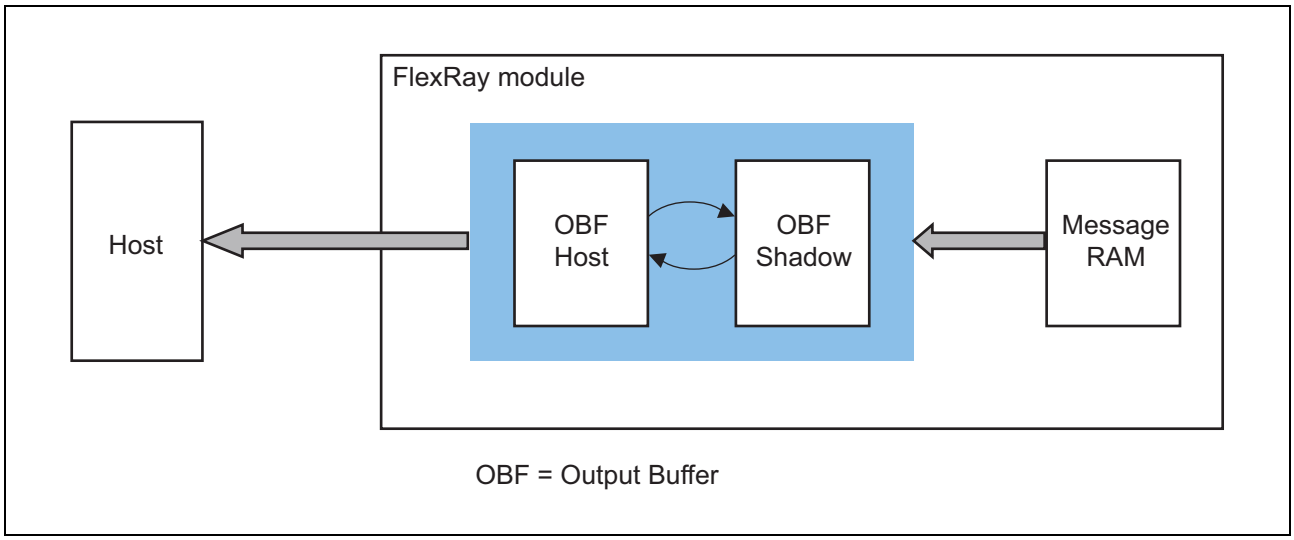


Figure 17.15 Double Buffer Structure Output Buffer

OBF host and OBF shadow as well as bits FLXA0FROBCM.RHSS, FLXA0FROBCM.RDSS, FLXA0FROBCM.RHSH, FLXA0FROBCM.RDSH and bits FLXA0FROBCR.OBRS, FLXA0FROBCR.OBRH are switched under control of bits FLXA0FROBCR.VIEW and FLXA0FROBCR.REQ.

Writing bit FLXA0FROBCR.REQ to ‘1’ copies bits FLXA0FROBCM.RHSS, FLXA0FROBCM.RDSS and bits FLXA0FROBCR.OBRS to an internal memory (see **Figure 17.16**).

After setting FLXA0FROBCR.REQ to ‘1’, FLXA0FROBCR.OBSYS is set to ‘1’, and the transfer of the message buffer selected by FLXA0FROBCR.OBRS from the message RAM to OBF shadow is started. After the transfer between the message RAM and OBF shadow has completed, the FLXA0FROBCR.OBSYS bit is set back to ‘0’. Bits FLXA0FROBCR.REQ and FLXA0FROBCR.VIEW can only be set to ‘1’ while FLXA0FROBCR.OBSYS is ‘0’.

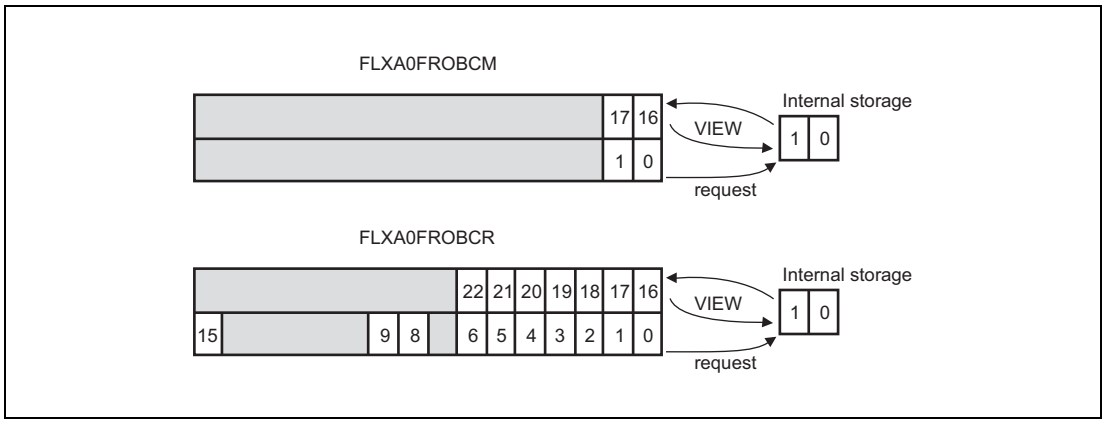


Figure 17.16 Switching of FLXA0FROBCM and FLXA0FROBCR Bits

OBF host and OBF shadow are switched by setting bit FLXA0FROBCR.VIEW to ‘1’ while bit FLXA0FROBCR.OBSYS is ‘0’ (see **Figure 17.15**).

In addition bits FLXA0FROBCR.OBRH and bits FLXA0FROBCM.RHSH, FLXA0FROBCM.RDSH are switched with the registers internal memory thus assuring that the message buffer number stored in FLXA0FROBCR.OBRH and the mask configuration stored in FLXA0FROBCM.RHSH, FLXA0FROBCM.RDSH matches the transferred data stored in OBF host (see **Figure 17.16**).

Now the host can read the transferred message buffer from OBF host while the message handler transfers the next message from the message RAM to OBF shadow.

If bits REQ and VIEW are set to ‘1’ with the same write access while FLXA0FROBSYS is ‘0’, FLXA0FROBSYS is automatically set to ‘1’ and OBF shadow and OBF host are switched. Additionally mask bits FLXA0FROBCM.RDSH and FLXA0FROBCM.RHSH are switched with the registers internal storage to keep them attached to the respective output buffer transfer. Afterwards FLXA0FROBCR.OBRS is copied to the register internal storage, mask bits FLXA0FROBCM.RDSS and FLXA0FROBCM.RHSS are copied to register FLXA0FROBCM internal memory, and the transfer of the selected message buffer from the message RAM to OBF shadow is started. While the transfer is ongoing the host can read the message buffer transferred by the previous transfer from OBF host. When the current transfer between message RAM and OBF shadow has completed, completion of the transfer is notified by setting FLXA0FROBCR.OBSYS back to ‘0’.

Example of an 8/16/32-bit host access to a single message buffer:

If a single message buffer has to be read out, two separate write accesses to FLXA0FROBCR.REQ and FLXA0FROBCR.VIEW are necessary:

- Wait until FLXA0FROBCR.OBSYS is reset
- Write output buffer command mask FLXA0FROBCM.RHSS, FLXA0FROBCM.RDSS
- Request transfer of message buffer to OBF shadow by writing FLXA0FROBCR.OBRS and FLXA0FROBCR.REQ (in case of and 8-bit host interface, FLXA0FROBCR.OBRS has to be written before FLXA0FROBCR.REQ).
- Wait until FLXA0FROBCR.OBSYS is reset
- Toggle OBF shadow and OBF host by writing FLXA0FROBCR.VIEW = ‘1’
- Read out transferred message buffer by reading FLXA0FRRDDSn, FLXA0FRRDHS1…3, and FLXA0FRMBS

Example of an 8/16/32-bit host access sequence:

Request transfer of 1st message buffer to OBF shadow

- Wait until FLXA0FROBCR.OBSYS is reset
- Write output buffer command mask FLXA0FROBCM.RHSS, FLXA0FROBCM.RDSS for 1st message buffer
- Request transfer of 1st message buffer to OBF shadow by writing FLXA0FROBCR.OBRS and FLXA0FROBCR.REQ (in case of an 8-bit host interface, FLXA0FROBCR.OBRS has to be written before FLXA0FROBCR.REQ).

Toggle OBF shadow and OBF host to read out 1st transferred message buffer and request transfer of 2nd message buffer:

- Wait until FLXA0FROBCR.OBSYS is reset to ‘0’

- Write output buffer command mask FLXA0FROBCM.RHSS, FLXA0FROBCM.RDSS for 2nd message buffer
- Toggle OBF shadow and OBF host and start transfer of 2nd message buffer to OBF shadow simultaneously by writing FLXA0FROBCR.OBRS of 2nd message buffer, FLXA0FROBCR.REQ, and FLXA0FROBCR.VIEW (in case of and 8-bit host interface, FLXA0FROBCR.OBRS has to be written before FLXA0FROBCR.REQ and FLXA0FROBCR.VIEW).
- Read out 1st transferred message buffer by reading FLXA0FRRDDSn, FLXA0FRRDHS1…3, and FLXA0FRMBS

…

Demand access to last requested message buffer without a request of another message buffer:

- Wait until FLXA0FROBCR.OBSYS is reset to ‘0’
- Demand access to last transferred message buffer by writing FLXA0FROBCR.VIEW
- Read out last transferred message buffer by reading FLXA0FRRDDSn, FLXA0FRRDHS1…3, and FLXA0FRMBS

Table 17.98 Assignment of FLXA0FROBCM Bits

Pos.	Access	Bit	Function
17	r	RDSH	Data Section Accessible
16	r	RHSH	Header Section Accessible
1	r/w	RDSS	Read Data Section Shadow
0	r/w	RHSS	Read Header Section Shadow

Table 17.99 Assignment of FLXA0FROBCR Bits

Pos.	Access	Bit	Function
22…16	r	OBRH	OBF Host Transfer Request Flag, number of message buffer accessible
15	r	OBSYS	OBF Busy Shadow Flag, signals ongoing transfer from message RAM to OBF shadow
9	r/w	REQ	Request transfer from message RAM to OBF shadow
8	r/w	VIEW	View OBF Shadow, swap OBF shadow and OBF host
6…0	r/w	OBRS	OBF Request Shadow, number of message buffer for next request

17.3.12.3 FlexRay Protocol Controller Access to Message RAM

The two temporary buffers (TBF A and TBF B) are used to buffer the data for transfer between the two FlexRay protocol controllers and the message RAM.

Each temporary buffer is build up as a double buffer, and able to store two complete FlexRay messages. There is always one buffer assigned to the corresponding protocol controller while the other one is accessible by the message handler.

If e.g. the message handler writes the next message to be sent to temporary buffer Tx, the FlexRay channel protocol controller can access temporary buffer Rx to store the message it is actually receiving. During transmission of the message stored in temporary buffer Tx, the message handler transfers the last received message stored in temporary buffer Rx to the message RAM (if it passes acceptance filtering) and updates the corresponding message buffer.

Data transfers between the temporary buffers and the shift registers of the FlexRay channel protocol controllers are done in words of 32 bit. This enables the use of a 32 bit shift register regardless of the length of the FlexRay messages.

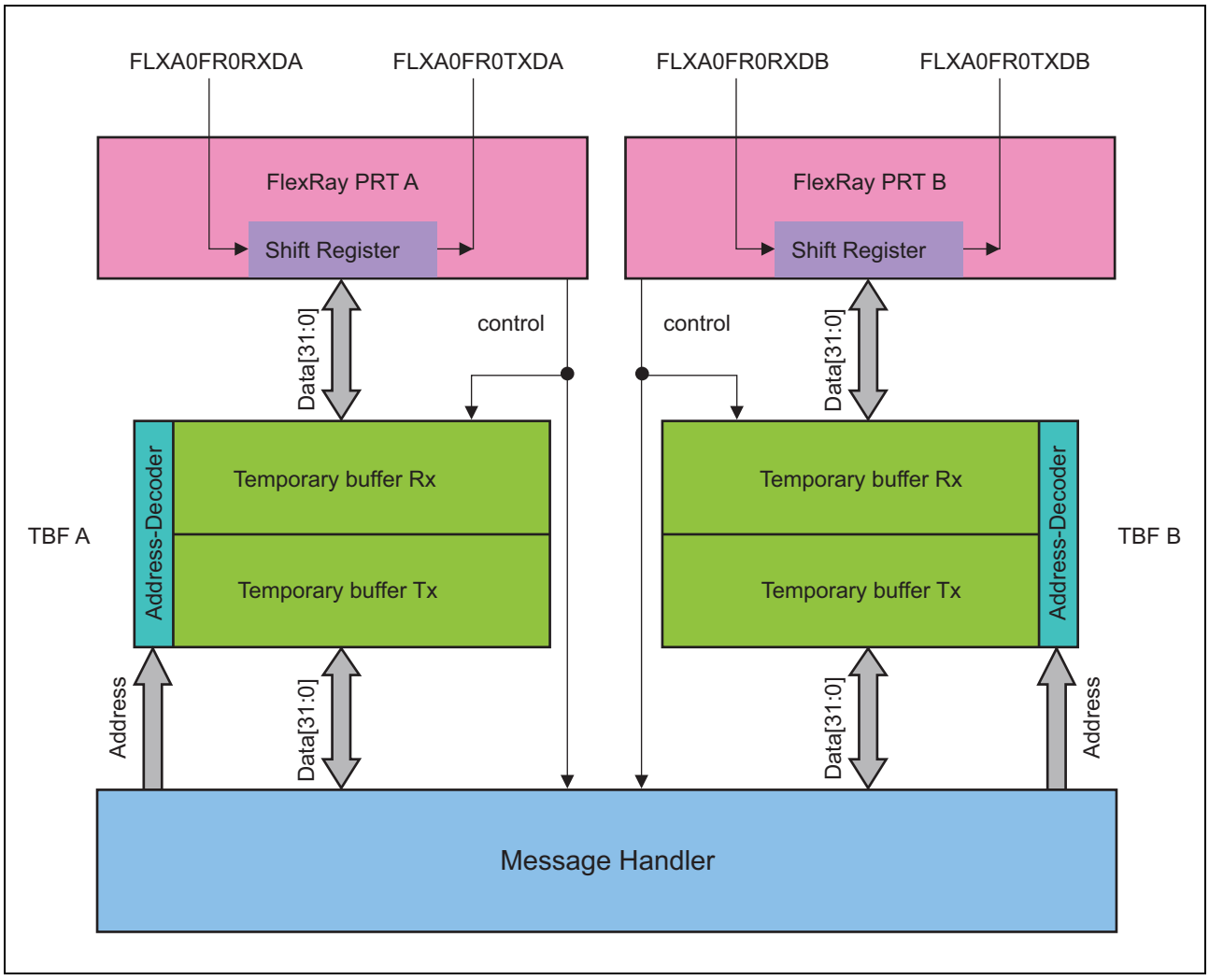


Figure 17.17 Access to Temporary Buffers

17.3.13 Message RAM

To avoid conflicts between host access to the message RAM and FlexRay message reception/ transmission, the host cannot directly access the message buffers in the message RAM. These accesses are handled via the input and output buffers. The Message RAM is able to store up to 128 message buffers depending on the configured payload length.

The message RAM is able to store up to 2048 32-bit words. To achieve the required flexibility with respect to different numbers of data bytes per FlexRay frame (0…254), the message RAM has a structure as shown in **Figure 17.18**.

When the message buffer of the data section to be allocated after the header partition is set as a reception buffer (by setting the FLXAnFRWRHS1.CFG bit to 0) or as a reception FIFO buffer, set an unused area of at least 32 bits at the start of the data section. In this case, the data partition can be started from the nth word in the message RAM where n calculated from (the setting of the FLXAnFRMRC.LCB[7:0] bits + 1) × 4 + 1.

When the message buffer of the data section to be allocated after the header partition is set as a transmission buffer (by setting the FLXAnFRWRHS1.CFG bit to 1), the data partition can be started from the nth word in the message RAM where n calculated from (the setting of the FLXAnFRMRC.LCB[7:0] bits + 1) × 4.

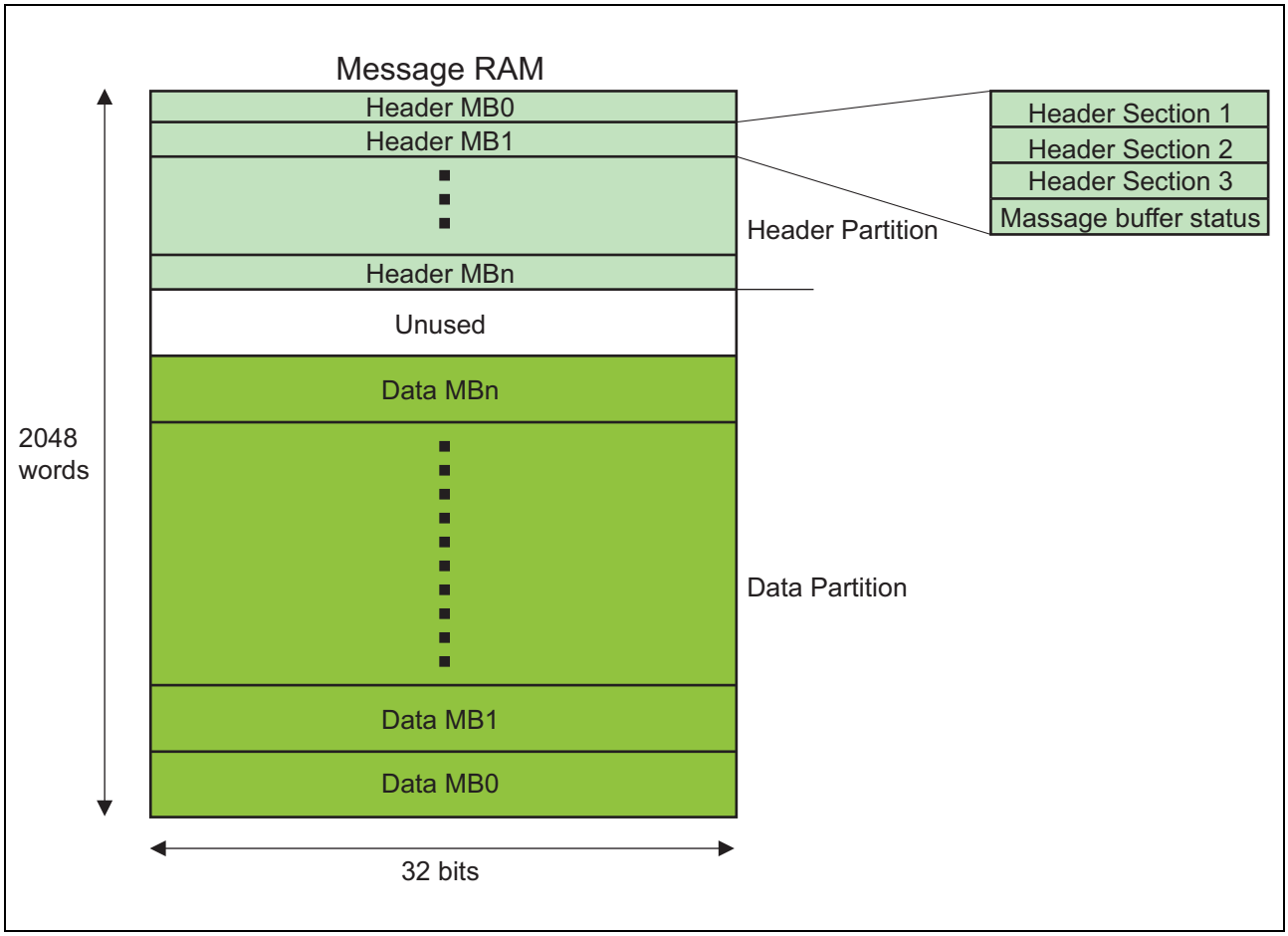


Figure 17.18 Configuration Example of Message Buffers in the Message RAM

Header Partition

Stores header sections of the configured message buffers:

- Supports a maximum of 128 message buffers
- Each message buffer has a header section of four 32-bit words
- Header 3 of each message buffer holds the 11-bit data pointer to the respective data section in the data partition

Data Partition

Flexible storage of data sections with different length. Some maximum values are:

- 30 message buffers with 254 byte data section each
- Or 56 message buffers with 128 byte data section each
- Or 128 message buffers with 48 byte data section each

NOTE

The total size of the header partition and data partition must be set less than 2048 32-bit words.

17.3.13.1 Header Partition

The elements used for configuration of a message buffer as well as the actual message buffer status are stored in the header partition of the message RAM as listed in **Table 17.100** below. Configuration of the header sections of the message buffers is done via IBF (FLXA0FRWRHS1…3). Read access to the header sections is done via OBF (FLXA0FRRDHS1…3 + FLXA0FRMBS). The data pointer has to be calculated to define the starting point of the data section for the corresponding message buffer in the data partition of the message RAM. The data pointer should not be modified during runtime. For message buffers belonging to the receive FIFO (re)configuration is possible in DEFAULT_CONFIG or CONFIG state only.

The header section of each message buffer occupies four 32-bit words in the header partition of the message RAM. The header of message buffer 0 starts with the first word in the message RAM.

For transmit buffers the Header CRC has to be calculated by the host.

The followings are updated from received valid data frames only:
Payload Length Received (PLR bit), Receive Cycle Count (RCC bit), Received on Channel Indicator (RCI bit), Startup Frame Indicator (SFI bit), Sync Frame Indicator (SYN bit), Null Frame Indicator (NFI bit), Payload Preamble Indicator (PPI bit), and Reserved Bit (RES).

Table 17.100 Header Section of a Message Buffer in the Message RAM

Bit Word	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0			MBI	TXM	PPIT	CFG	CH			Cycle Code												Frame ID										
1		Payload Length Received								Payload length Configured												Tx Buffer: Header CRC Configured Rx Buffer: Header CRC Received										
2			RES	PPI	NFI	SYN	SFI	RCI			Receive Cycle Count											Data Pointer										
3			RESS	PPIS	NFIS	SYNS	SFIS	RCIS			Cycle Count Status						FTB	FTA		MLST	ESB	ESA	TCIB	TCIA	SVOB	SVOA	CEOB	CEOA	SEOB	SEOA	VFRB	VFRA
...	...																															
...	...																															

Frame Configuration
Filter Configuration
Message Buffer Control
Message RAM Configuration
Updated from received Data Frame
Message Buffer Status (MBS)
unused

(1) Header section 1 (word 0)

Write access via FLXA0FRWRHS1, read access via FLXA0FRRDHS1:

- Frame ID
  - Slot counter filtering configuration
- Cycle Code
  - Cycle counter filtering configuration
- CH
  - Channel filtering configuration
- CFG
  - Message buffer direction configuration: reception or transmission
- PPIT
  - Payload Preamble Indicator Transmit
- TXM
  - Transmit mode configuration: single-shot or continuous
- MBI
  - Message buffer reception/transmission interrupt enable

(2) Header section 2 (word 1)

Write access via FLXA0FRWRHS2, read access via FLXA0FRRDHS2:

- Header CRC
  - Transmit Buffer: Configured by the host (calculated from frame header)
  - Receive Buffer: Updated from received frame
- Payload Length Configured
  - Length of data section (2-byte words) as configured by the host
- Payload Length Received
  - Length of payload segment (2-byte words) stored from received frame

(3) Header section 3 (word 2)

Write access via FLXA0FRWRHS3, read access via FLXA0FRRDHS3:

- Data Pointer
  - Pointer to the beginning of the corresponding data section in the data partition

Read access via FLXA0FRRDHS3, valid for receive buffers only, updated from received frames:

- Receive Cycle Count
  - Cycle count from received frame
- RCI
  - Received on Channel Indicator

- SFI
  - Startup Frame Indicator
- SYN
  - Sync Frame Indicator
- NFI
  - Null Frame Indicator
- PPI
  - Payload Preamble Indicator
- RES
  - Reserved bit

(4) Message Buffer Status FLXA0FRMBS (word 3)

Read access via FLXA0FRMBS, updated by the CC at the end of the configured slot.

- VFRA
  - Valid Frame Received on channel A
- VFRB
  - Valid Frame Received on channel B
- SEOA
  - Syntax Error Observed on channel A
- SEOB
  - Syntax Error Observed on channel B
- CEOA
  - Content Error Observed on channel A
- CEOB
  - Content Error Observed on channel B
- SVOA
  - Slot boundary Violation Observed on channel A
- SVOB
  - Slot boundary Violation Observed on channel B
- TCIA
  - Transmission Conflict Indication channel A
- TCIB
  - Transmission Conflict Indication channel B
- ESA
  - Empty Slot Channel A

- ESB
  - Empty Slot Channel B
- MLST
  - Message Lost
- FTA
  - Frame Transmitted on Channel A
- FTB
  - Frame Transmitted on Channel B
- Cycle Count Status
  - Actual cycle count when status was updated
- RCIS
  - Received on Channel Indicator Status
- SFIS
  - Startup Frame Indicator Status
- SYNS
  - Sync Frame Indicator Status
- NFIS
  - Null Frame Indicator Status
- PPIS
  - Payload Preamble Indicator Status
- RESS
  - Reserved bit Status

17.3.13.2 Data Partition

The data partition of the message RAM stores the data sections of the message buffers configured for reception/transmission as defined in the header partition. The number of data bytes for each message buffer can vary from 0 to 254. To optimize the data transfer between the shift registers of the two FlexRay protocol controllers and the message RAM as well as between the host interface and the message RAM, the physical width of the message RAM is set to 4 bytes.

The data partition starts after the last word of the header partition. When configuring the message buffers in the message RAM the programmer has to assure that the data pointers point to addresses within the data partition. **Table 17.101** below shows an example how the data sections of the configured message buffers can be stored in the data partition of the message RAM.

The beginning and the end of a message buffer's data section is determined by the data pointer and the payload length configured in the message buffer's header section, respectively. This enables a flexible usage of the available RAM space for storage of message buffers with different data length.

If the size of the data section is an odd number of 2-byte words, the remaining 16 bits in the last 32-bit word are unused (see **Table 17.101** below).

Table 17.101 Example for Structure of the Data Partition in the Message RAM

Bit Word	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
...	Unused								Unused								Unused								Unused							
...	Unused								Unused								Unused								Unused							
...	MBn data 3								MBn data 2								MBn data 1								MBn data 0							
...	...								...								...								...							
...	...								...								...								...							
...	MBn data (m)								MBn data (m-1)								MBn data (m-2)								MBn data (m-3)							
...	...								...								...								...							
...	...								...								...								...							
...	...								...								...								...							
...	MB1 data 3								MB1 data 2								MB1 data 1								MB1 data 0							
...	...								...								...								...							
2046	MB0 data 3								MB0 data 2								MB0 data 1								MB0 data 0							
2047	Unused								Unused								MB0 data 5								MB0 data 4							

17.3.13.3 Message Data Integrity Check

A data integrity checking mechanism is implemented in the FlexRay core to assure the integrity of the data stored in the related RAM. Each RAM has a checksum generator and checker attached as shown in **Figure 17.19**.

When data is written to a RAM, the local checksum generator generates the checksum. The checksum is stored together with the respective data word. The checksum is checked each time a data word is read from a RAM.

If a checksum error is detected, the respective access error flag is set. The access error flags FLXA0FRMHDS.AMR, FLXA0FRMHDS.ATBF1, FLXA0FRMHDS.ATBF2 and the faulty message buffer indicators FLXA0FRMHDS.FMBD, FLXA0FRMHDS.MFMB, FLXA0FRMHDS.FMB are located in the FlexRay message handler status register. These single access error flags control the error interrupt flag FLXA0FREIR.AERR.

Figure 17.19 shows the data paths between the input buffer, temporary buffer and message RAM.

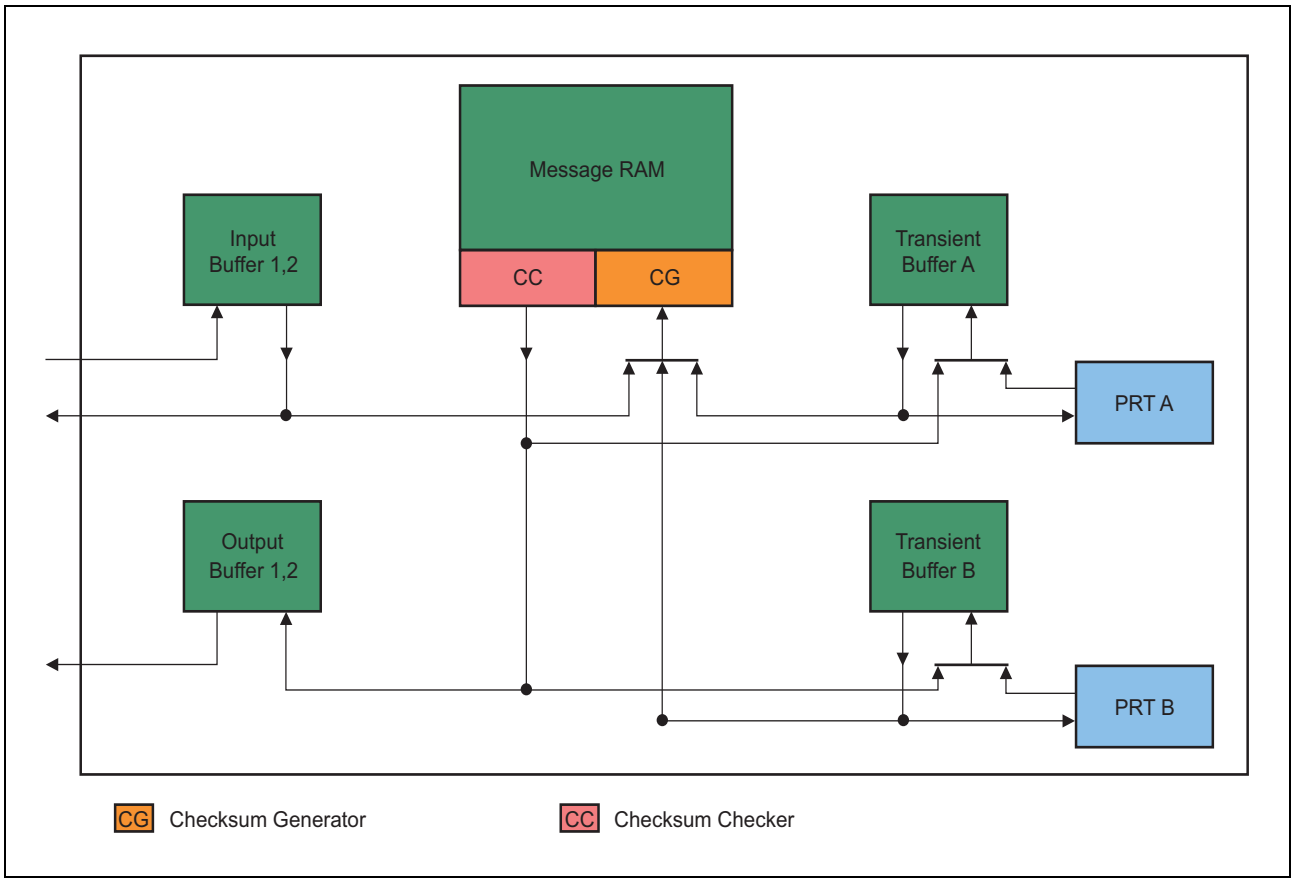


Figure 17.19 Checksum Generation and Check

When an access error has been detected the following actions will be performed:

In all cases:

- The respective access error flag in FLXA0FRMHDS register is set
- The access error flag FLXA0FREIR.AERR is set and, if enabled, a module interrupt to the host will be generated.

Additionally in specific cases:

(1) Access error during data transfer from input buffer 1 or 2 to message RAM when reading header section of respective message buffer from message RAM:

- FLXA0FRMHDS.AMR is set.
- FLXA0FRMHDS.FMBD bit is set to indicate that there is an error in the message buffer to which FLXA0FRMHDS.FMB points.
- FLXA0FRMHDS.FMB indicates the number of the message buffer which has an error.
- The data section of the corresponding message buffer is not updated.
- Transmit buffer: Transmission request for the corresponding message buffer is not set.

(2) Access error during scan of header sections in message RAM:

- FLXA0FRMHDS.AMR is set.
- FLXA0FRMHDS.FMBD bit is set to indicate that there is an error in the message buffer to which FLXA0FRMHDS.FMB points.
- FLXA0FRMHDS.FMB indicates the number of the message buffer which has an error.
- The message buffer is ignored (skipped).

(3) Access error during data transfer from message RAM to temporary buffer 1 or 2:

- FLXA0FRMHDS.AMR is set.
- FLXA0FRMHDS.FMBD bit is set to indicate that there is an error in the message buffer to which FLXA0FRMHDS.FMB points.
- FLXA0FRMHDS.FMB indicates the number of the message buffer which has an error.
- Frame is not transmitted. After the frame CRC is set to 0, the frame which is being transmitted is invalidated.

(4) Access error during data transfer from temporary buffer 1 or 2 to message RAM when reading header section of respective message buffer from message RAM:

- FLXA0FRMHDS.AMR is set.
- FLXA0FRMHDS.FMBD bit is set to indicate that there is an error in the message buffer to which FLXA0FRMHDS.FMB points.
- FLXA0FRMHDS.FMB indicates the number of the message buffer which has an error.
- The data section of the corresponding message buffer is not updated.

(5) Access error during data transfer from message RAM to output buffer:

- The access error flag FLXA0FRMHDS.AMR is set.
- FLXA0FRMHDS.FMBD bit is set to indicate that there is an error in the message buffer to which FLXA0FRMHDS.FMB points.
- FLXA0FRMHDS.FMB indicates the number of the message buffer which has an error.

(6) Access error during a data transfer from temporary buffer 1 or 2 to protocol controller 1 or 2:

- FLXA0FRMHDS.ATBF1 and FLXA0FRMHDS.ATBF2 bits are set.
- After the frame CRC is set to 0, the frame which is being transmitted is invalidated.

(7) Access error during data transfer from temporary buffer 1 or 2 to message RAM when reading temporary buffer 1 or 2:

- FLXA0FRMHDS.ATBF1 and FLXA0FRMHDS.ATBF2 bits are set.
- FLXA0FRMHDS.FMBD bit is set to indicate that there is an error in the message buffer to which FLXA0FRMHDS.FMB points.
- FLXA0FRMHDS.FMB indicates the number of the message buffer which has an error.

(8) Access error during data read of temporary buffer RAM 1 or 2:

- When an access error occurs while the message handler read a frame with network management information (PPI = '1') from temporary buffer RAM 1 or 2, the corresponding network management vector registers FLXA0FRNMV1 to 3 are not updated from this frame.

17.3.13.4 Host Handling of Access Errors

Access error caused by temporary bit flips can be fixed by:

(1) Self-healing

Access errors located in the data section of message RAM, temporary buffer RAM A or temporary buffer RAM B is overwritten with the next write access to the disturbed bit(s) caused by host access or by FlexRay communication.

(2) CLEAR_RAMS Command

The POC command CLEAR_RAMS initializes the message RAM to zero, when called in the DEFAULT_CONFIG or CONFIG state.

(3) Temporary Unlocking of Header Section

An access error in the header section of a locked message buffer can be fixed by a transfer from the input buffer to the locked buffer header section. For this transfer, the write access to the FLXA0FRIBCR (specifying the message buffer number) must be immediately preceded by the unlock sequence normally used to leave CONFIG state (see **Section 17.2.3.1, FLXA0FRLCK — FlexRay Lock Register**).

For that single transfer the message buffer header is unlocked, regardless whether it belongs to the FIFO or whether its locking is controlled by FLXA0FRMRC.SEC, and will be updated with new data.

17.3.14 Interrupts

In general, interrupts provide a close link to the protocol timing as they are triggered almost immediately when an error or status change is detected by the CC, a frame is received or transmitted, a configured timer interrupt is activated, or a stop watch event occurred. This enables the host to react very quickly on specific error conditions, status changes, or timer events. On the other hand, too many interrupts can cause the host to miss deadlines required by the application. Therefore the CC supports enable and disable controls for each individual interrupt source separately.

An interrupt may be triggered when

- An error was detected
- A status flag is set to '1'
- A timer reaches a configured value
- A message transfer from input buffer to message RAM or from message ram to output buffer has completed
- A message transfer from the local RAM/global RAM to message RAM or from message RAM to local RAM/global RAM has completed
- A stop watch event occurred

Tracking status and generating interrupts when a status change or an error occurs are two independent tasks. Regardless of whether an interrupt is enabled or not, the corresponding status is tracked and indicated by the CC. The host has access to the actual status and error information by reading registers FLXA0FREIR, FLXA0FRSIR, FLXA0FROS, FLXA0FROTS and FLXA0FRITS.

The general purpose interrupt lines to the host, FlexRay Interrupt 0, FlexRay Interrupt 1, are controlled by the enabled interrupts in FLXA0FREIES and FLXA0FRSIES. In addition each of the two interrupt lines can be enabled or disabled separately by programming bit FLXA0FRILE.EINT0 and FLXA0FRILE.EINT1.

The input data transfer interrupt lines to the host, FlexRay input queue empty interrupt, FlexRay input queue full interrupt, are controlled by the enabled interrupts in FLXA0FRITS. In addition each of the input data transfer interrupts can be enabled or disabled separately by programming the related bits in FLXA0FRITC.

The output data transfer interrupt lines to the host, FlexRay FIFO transfer warning interrupt, FlexRay output transfer warning interrupt, FlexRay FIFO transfer interrupt, FlexRay output transfer interrupt, are controlled by the enabled interrupts in FLXA0FROTS. In addition each of the output data transfer interrupts can be enabled or disabled separately by programming the related bits in FLXA0FROTC.

The three timer interrupts lines to the host are controlled by the enabled interrupts in FLXA0FROS. In addition, each of the interrupt lines can be enabled or disabled separately by programming bit FLXA0FROC.T0IE, FLXA0FROC.T1IE and FLXA0FROC.T2IE.

When a transfer between IBF/OBF and the message RAM has completed, bit FLXA0FRSIR.TIBC or FLXA0FRSIR.TOBC is set to '1'.

17.3.15 Assignment of FlexRay Configuration Parameters

Table 17.102 FlexRay Configuration Parameters (1/2)

Parameter	Bit(field)
pKeySlotusedForStartup	FLXA0FRSUCC1.TXST
pKeySlotUsedForSync	FLXA0FRSUCC1.TXSY
gColdStartAttempts	FLXA0FRSUCC1.CSA
pAllowPassiveToActive	FLXA0FRSUCC1.PTA
pWakeupChannel	FLXA0FRSUCC1.WUCS
pSingleSlotEnabled	FLXA0FRSUCC1.TSM
pAllowHaltDueToClock	FLXA0FRSUCC1.HCSE
pChannels	FLXA0FRSUCC1.CCH
pdListenTimeOut	FLXA0FRSUCC2.LT
gListenNoise	FLXA0FRSUCC2.LTN
gMaxWithoutClockCorrectionPassive	FLXA0FRSUCC3.WCP
gMaxWithoutClockCorrectionFatal	FLXA0FRSUCC3.WCF
gNetworkManagementVectorLength	FLXA0FRNEMC.NML
gdTSSTransmitter	FLXA0FRPRTC1.TSST
gdCASRxLowMax	FLXA0FRPRTC1.CASM
gdSampleClockPeriod	FLXA0FRPRTC1.BRP
pSamplesPerMicrotick	FLXA0FRPRTC1.BRP
gdWakeupSymbolRxWindow	FLXA0FRPRTC1.RXW
pWakeupPattern	FLXA0FRPRTC1.RWP
gdWakeupSymbolRxIdle	FLXA0FRPRTC2.RXI
gdWakeupSymbolRxLow	FLXA0FRPRTC2.RXL
gdWakeupSymbolTxIdle	FLXA0FRPRTC2.TXI
gdWakeupSymbolTxLow	FLXA0FRPRTC2.TXL
gPayloadLengthStatic	FLXA0FRMHDC.SFDL
pLatestTx	FLXA0FRMHDC.SLT
pMicroPerCycle	FLXA0FRGTUC1.UT
gMacroPerCycle	FLXA0FRGTUC2.MPC
gSyncNodeMax	FLXA0FRGTUC2.SNM
pMicroInitialOffset[A]	FLXA0FRGTUC3.UIOA
pMicroInitialOffset[B]	FLXA0FRGTUC3.UIOB
pMacroInitialOffset[A]	FLXA0FRGTUC3.MIOA
pMacroInitialOffset[B]	FLXA0FRGTUC3.MIOB
gdNIT	FLXA0FRGTUC4.NIT
gOffsetCorrectionStart	FLXA0FRGTUC4.OCS
pDelayCompensation[A]	FLXA0FRGTUC5.DCA
pDelayCompensation[B]	FLXA0FRGTUC5.DCB
pClusterDriftDamping	FLXA0FRGTUC5.CDD
pDecodingCorrection	FLXA0FRGTUC5.DEC
pdAcceptedStartupRange	FLXA0FRGTUC6.ASR
pdMaxDrift	FLXA0FRGTUC6.MOD
gdStaticSlot	FLXA0FRGTUC7.SSL
gNumberOfStaticSlots	FLXA0FRGTUC7.NSS

Table 17.102 FlexRay Configuration Parameters (2/2)

Parameter	Bit(field)
gdMinislot	FLXA0FRGTUC8.MSL
gNumberOfMinislots	FLXA0FRGTUC8.NMS
gdActionPointOffset	FLXA0FRGTUC9.APO
gdMinislotActionPointOffset	FLXA0FRGTUC9.MAPO
gdDynamicSlotIdlePhase	FLXA0FRGTUC9.DSI
pOffsetCorrectionOut	FLXA0FRGTUC10.MOC
pRateCorrectionOut	FLXA0FRGTUC10.MRC
pExternOffsetCorrection	FLXA0FRGTUC11.EOC
pExternRateCorrection	FLXA0FRGTUC11.ERC

17.3.16 Usage of Data Transfer

A mechanism is implemented to allow storage of FlexRay messages directly into the local RAM/global RAM (user RAM) and have transfers between the FlexRay internal message RAM and the local RAM/global RAM and vice versa with minimum CPU support. The data in the local RAM/global RAM should be indexed by data structure pointers located in data pointer tables stored in the local RAM/global RAM.

Data transfer from the local RAM/global RAM to the FlexRay internal message RAM (input transfer) needs to be initiated by the application. These transfers can be used to configure message buffers or to update transmit data.

A data transfer from the FlexRay internal message RAM to the local RAM/global RAM (output transfer) is initiated automatically by a reception into a receive message buffer or FlexRay internal FIFO or by a change in the slot status. It can be initiated also by a specific user transfer request.

The input and output data transfer can be activated independently. When the input data transfer is activated the application should not directly access message buffers using the FlexRay input buffer. When the output data transfer is activated the application cannot directly access message buffers using the FlexRay output buffer.

17.3.16.1 Input Data Transfer

When the automatic input data transfer function is enabled, committed input data structures are transferred from the local RAM/global RAM to the FlexRay internal message RAM with minimum CPU support.

(1) Activation and deactivation

The input data transfer function should be activated before usage. The activation of the input transfer handler initializes the input queue put index (FLXA0FRITS.IPIDX) and get index (FLXA0FRITS.IGIDX) to zero. In addition, set the interrupt status flags in the FLXA0FRITS register (IQEIS and IQFIS) to '0'.

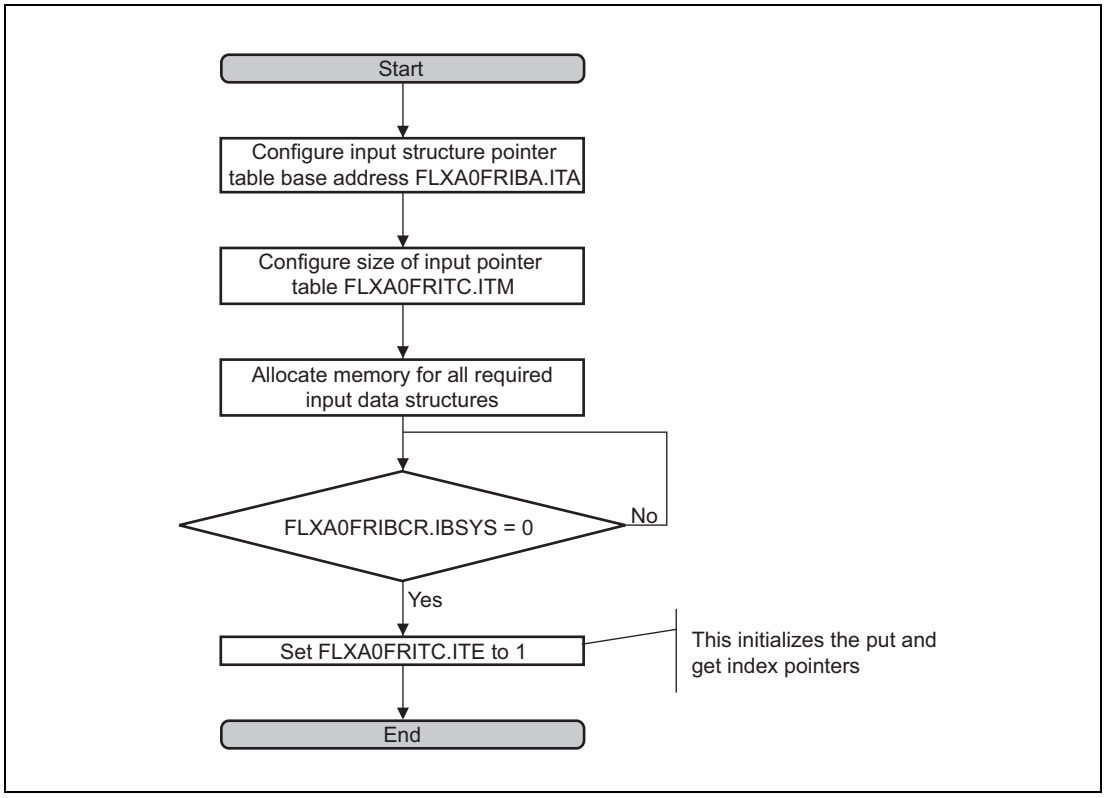


Figure 17.20 Input Transfer Enable Flow

A deactivation request of the input transfer function can be made at any time. The input queue put index and the input queue status are maintained independently from the input transfer function state.

Before the transfer function is disabled (status indicated by FLXA0FRITS.ITS), user requested input transfers and all committed input transfers will be completed.

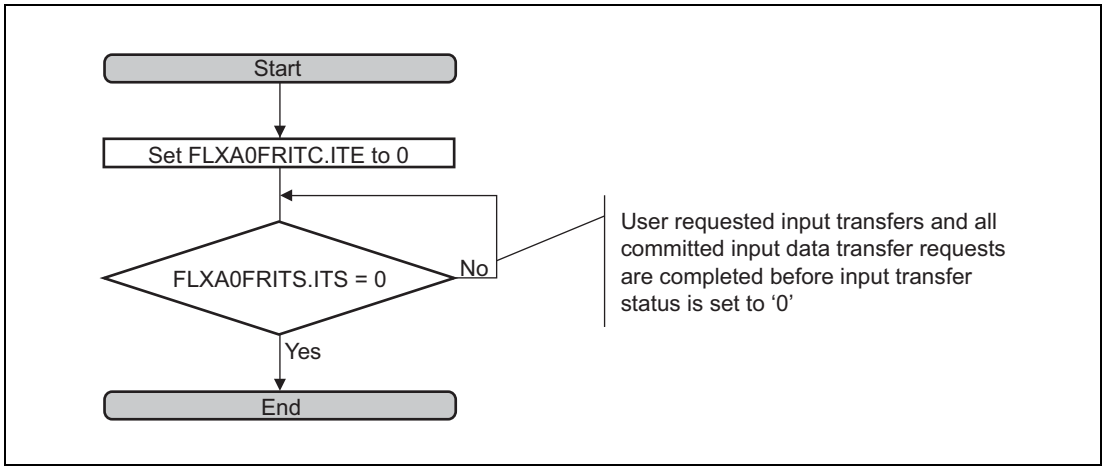


Figure 17.21 Input Transfer Disable Flow

(2) Input data structure

The application has to reserve a location in the local RAM/global RAM to provide the content for message buffer configuration (input data structure).

The location of this input data structure needs to also be defined by an input data structure pointer located in the local RAM/global RAM.

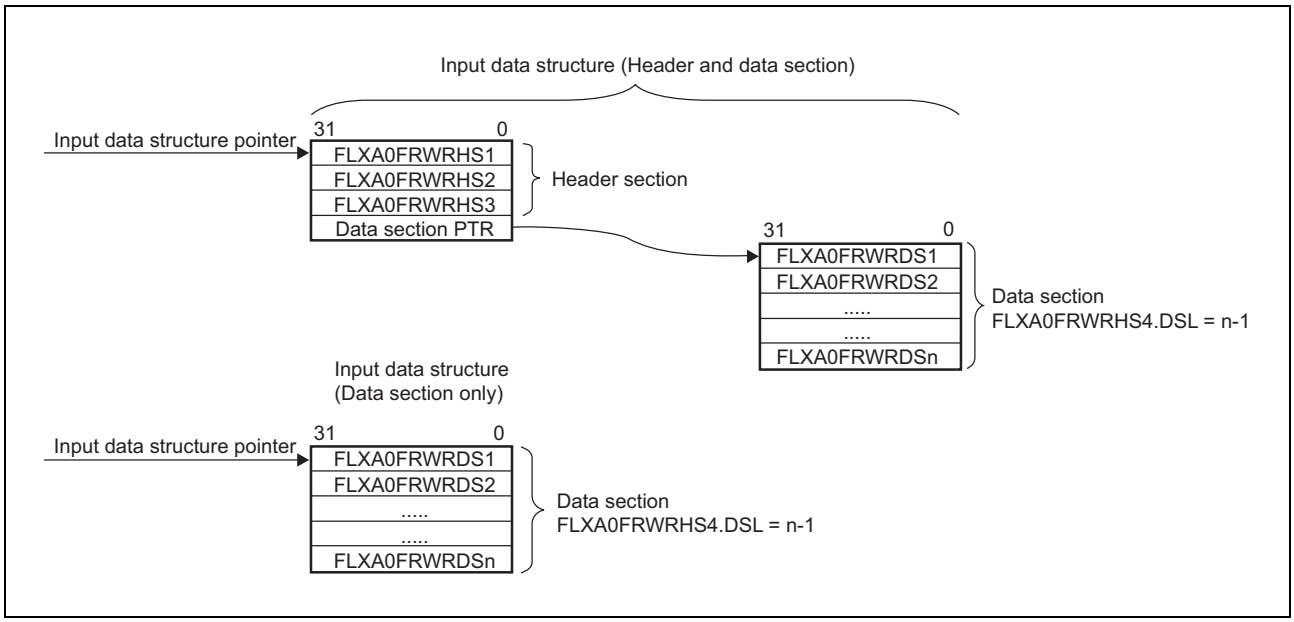


Figure 17.22 Input Data Structure

In general the input data structure consists of two sections, the header and the data section.

The header section consists of FLXA0FRWRHS1, FLXA0FRWRHS2, FLXA0FRWRHS3 and the data section pointer.

For bit alignment and bit function in the header section, see **Section 17.3.13.1, Header Partition**.

Depending on the settings in the control field (FLXA0FRWRHS4) located in the input pointer table, the data structure pointer is a reference to the address of FLXA0FRWRHS1 or FLXA0FRWRDS1. The data structure pointer has to be aligned to a 32 bit address.

If the bit LHS in the address related to FLXA0FRWRHS4 is set to ‘1’ it is required to provide a valid header section. In this case FLXA0FRWRHS1 is the first element of the data structure.

If the bit LHS in the address related to FLXA0FRWRHS4 is set to ‘0’ a header section is not required. In this case FLXA0FRWRDS1 is the first element of the data structure.

If the bit LDS in the address related to FLXA0FRWRHS4 is set to ‘1’ it is required to provide a valid data section. The pointer to the data section is a reference to the address of the first payload long word (FLXA0FRWRDS1) and has to be aligned to a 32 bit address.

If the bit LDS in the address related to FLXA0FRWRHS4 is set to '0' a data section is not required. The data section pointer is not evaluated by the input handler.

The byte order for the FlexRay payload data in the input data structure is determined by FLXA0FROC.BEC. For information about the payload data alignment within the data section, refer to **Section 17.3.13.2, Data Partition** and **Section 17.3.17, Byte Alignment**.

The length of the data section and the size to be allocated in the local RAM/global RAM depend on the configuration of the bits DSL in the address related to FLXA0FRWRHS4.

For the transfer into the FlexRay core internal message RAM the number of 16 bit words configured by FLXA0FRWRHS2.PLC is used. The application has to ensure that a proper number of data words are provided in the local RAM/global RAM. In case the buffer is configured by FLXA0FRWRHS2.PLC to hold an odd payload length, the application has to write zero to the last 16 bit of the payload section to ensure that the padding data is all zero.

(3) Input pointer table

To transfer data from the input data structures located in the local RAM/global RAM to the FlexRay internal message RAM, the related input data structure pointer and control field need to be added to the input pointer table which is located in the local RAM/global RAM.

The location of the first element of this table is identified by the input pointer table base address (FLXA0FRIBA.ITA). This base address has to be aligned to a 32 bit address.

The maximum number of input requests that can be queued is defined by the Input queue Table Max register (FLXA0FRITC.ITM).

Each Input pointer table entry requires two long words. The required address range of the input pointer table for the queued transfer requests can be calculated by

Input pointer table size (byte) = (((FLXA0FRITC.ITM + 1) × 2) × 4

Equation 1

The input pointer entry for the user requested input transfer should be added after the end of the input pointer table.

The pointer table index related to this entry and hence the number to be written to FLXA0FRUIR.UIDX, is FLXA0FRITC.ITM+1. The address in the input pointer table related to the user requested input transfer (user input address) can be calculated by

User input address = FLXA0FRIBA.ITA + Input pointer table size

Equation 2

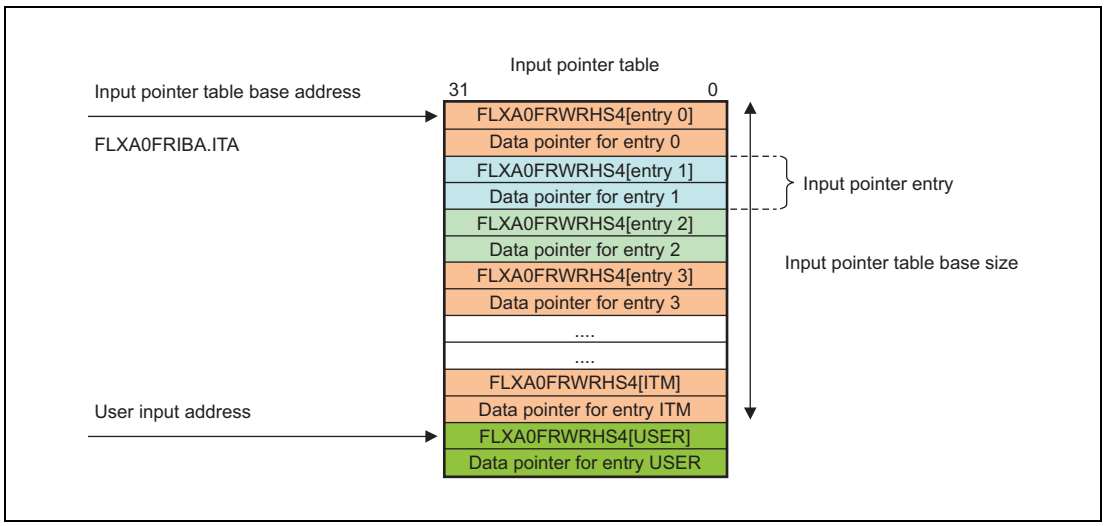


Figure 17.23 Input Pointer Table

The input pointer table holds the control field FLXA0FRWRHS4 and the pointers to the local RAM/ global RAM location where the message buffer content (header section and/or data section) is stored.

The application has to write FLXA0FRWRHS4 and the input data structure pointer at the addresses in the input pointer table related to the put index position before a transfer request is initiated.

FLXA0FRWRHS4:

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	DSL[5:0]					
Value after reset																
R/W																

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	INV	STR	LDS	LHS	—	IMBNR[6:0]						
Value after reset																
R/W																

Table 17.103 FLXA0FRWRHS4 Register Contents

Bit Position	Bit Name	Function
31 to 22	—	Reserved The read value is undefined. When writing, always write 0.
21 to 16	DSL	Data Section Length Specifies the length of the data section in terms of 32 bit values.
15 to 12	—	Reserved The read value is undefined. When writing, always write 0.
11	INV	Invalidate Entry 0: The data structure is valid and will be transferred to the FlexRay internal message RAM. 1: The data structure is invalid. FlexRay internal message RAM is not updated using this input pointer entry.
10	STR	Set Transmission Request 0: The bit FLXA0FRTXRQx.TXR for the message buffer selected by the bits IMBNR is set to ‘0’. No data from this message buffer is transmitted. 1: The bit FLXA0FRTXRQx.TXR for the message buffer selected by the bits IMBNR is set to ‘1’ to release the message buffer for transmission The application should not set the bit STR to ‘1’ for receive buffers.
9	LDS	Load Data Section 0: No update of data section 1: Data section for the message buffer selected by the bits IMBNR is updated.
8	LHS	Load Header Section 0: No update of header section 1: Header section for the message buffer selected by the bits IMBNR is updated.
7	—	Reserved The read value is undefined. When writing, always write 0.
6 to 0	IMBNR	Message Buffer Number Update Select the target message buffer number in the FlexRay internal message RAM for transfer.

Note that the LHS bit cannot be set for protected message buffers.

The bit LDS defines if the data section of the message buffer selected by the bits IMBNR is updated.

If LDS is set to ‘1’, (DSL + 1) 32-bit words of payload data are transferred from the local RAM/global RAM to the message buffer selected by the bits IMBNR.

If LDS is set to ‘0’, no payload data is transferred from the local RAM/global RAM.

Note that the payload transferred is independent from the configured payload length (bits PLC in the address related to FLXA0FRWRHS2).

The bit INV can be used to invalidate a transmitted data structure. This bit should be only used to cancel the transfer of committed data structures when the input queue is halted (see **Section 17.3.16.1 (5), Halting the input queue**).

When this bit is set to ‘1’, the message buffer number IMBNR is not updated. When the bit is set to '0', the message buffer number IMBNR is updated.

(4) Transfer function of input data structure

To use the input data structure transfer function, the input transfer has to be activated (see **Section 17.3.16.1 (1), Activation and deactivation**). The activation process requires the setup of the input pointer table (see **Section 17.3.16.1 (3), Input pointer table**) in order to specify the source location (input data structures) for the data structures to be transferred. When the input transfer is enabled the get index pointer is initialized to zero.

All FlexRay internal message buffers can be updated using the input transfer queue which is built in the input pointer table. The application has to write the pointer and control field (table entry) to the data structure to be transferred into the input pointer table. For that purpose the application has to maintain a put index for the input pointer table that indicates where the pointer has to be written to.

To transmit this table entry to the input handler, the application has to write the target message buffer number to the input queue control register (FLXA0FRIQC.IMBNR). Afterwards the application increments the application internal put index.

By writing to the input queue control register the data available flag (FLXA0FRDAi.DA[IMBNR]) is automatically set to ‘1’. The input transfer handler also maintains the put index pointer in the status register (FLXA0FRITS.IPIDX).

In case the input queue is full (number of queued input transfer requests is equal to the input queue table size), FLXA0FRITS.IQFP and FLXA0FRITS.IQFIS are set to ‘1’. The input queue full condition pending flag (FLXA0FRITS.IQFP) changes from ‘1’ to ‘0’ when there are entries in the input queue available, whereby the input queue full interrupt status flag (FLXA0FRITS.IQFIS) needs to be cleared by the application.

The application cannot make any further write access to the bit IMBNR in the FLXA0FRIQC register as long as the bit IQFP in the FLXA0FRITS register is ‘1’.

In case the input queue becomes empty (number of queued input transfer requests changes to zero) FLXA0FRITS.IQEIS is set to ‘1’.

The input queue empty interrupt status flag (FLXA0FRITS.IQEIS) needs to be cleared by the application.

The transfer of the input data structures to the FlexRay message RAM is controlled by a get index pointer which is handled inside the FlexRay module and flagged in FLXA0FRITS.IGIDX. Note that the index is referring to the input entry, but not the address offset in the input pointer table.

If the input queue is not empty, the transfer handler reads the input pointer table entry of the transfer queue and starts the transfer of the input data structure from the address the input pointer is referring to. When all required data words are transferred to the FlexRay module, the data available flag for the transferred message buffer number is set to ‘0’ and the get index in the transfer handler is incremented by one.

In case of an invalidated data structure (see **Section 17.3.16.1 (5), Halting the input queue**) no FlexRay internal message buffer is updated and the related data available flag is automatically set to ‘0’. The change of the data available flag can be used to confirm the cancellation a transmit request.

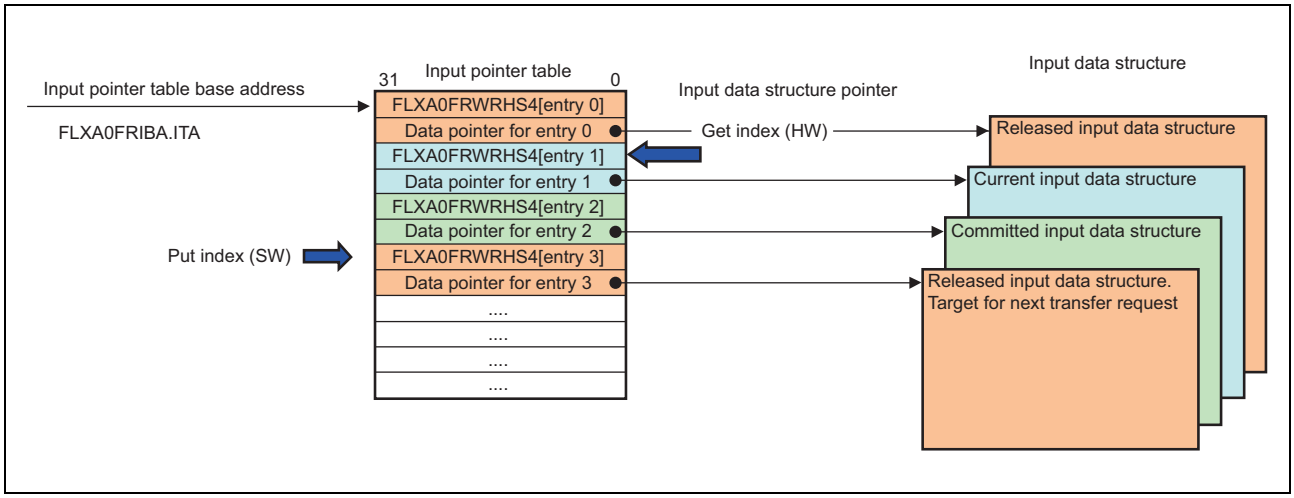


Figure 17.24 Input Pointer Table

Receive message buffers can be also configured using the input data transfer by setting up the required header sections and mark only the header section (FLXA0FRWRHS4.LDS = '0', FLXA0FRWRHS4.LHS = '1') to be updated in the FlexRay module.

The C-code below should be handled as an example of how the input queue could be handled in SW.

```
// Add the message block 'blk' into the FlexRay Tx queue
// If leng = -1, HS and DS is updated, else only DS
// Return 1, if the queue is full
int flex_add_transmit_block(int mbnr, int setTxReq, int leng,
t_is_block *blk)
{
    int mode;

    //ckeck if resource is free
    if (FRITS.IQFP != 0) return 1;

    mode  = (setTxReq << 10) + 0x20; /*load DS */
    if (leng == -1) {
        leng = (blk->hs2.plc+1) / 2; /* number of 32 bit word */
        mode |= 0x10; /* load HS + DS */
    }

    //add to input queue
    flex_tx_table[flex_tx_pidx*2+0] = blk;/* add the pointer */
    flex_tx_table[flex_tx_pidx*2+1] = (leng << 16) +mode + mbnr;

    //increment SW put index
    if (++flex_input_pidx == FLEX_INPUT_TABLE_MAX) flex_input_pidx = 0;

    //activate transmission
    FRIQC = mbnr;

    //done
    return 0;
}
```

(5) Halting the input queue

Committed data structures cannot be removed, but can be invalidated or updated when the input queue is halted.

To cancel data structures already committed to the input queue, the queue can be halted by writing a '1' to FLXA0FRITC.IQHR.

After the ongoing input transfer has been completed the queue is halted and FLXA0FRITS.IQH changes from '0' to '1'.

To invalidate an entry of the input queue, FLXA0FRWRHS4.INV has to be set to '1'. All other bits in FWRHS4 cannot be changed.

Following flow shall be used to analyze whether a transmitted message has been already transferred to the FlexRay internal message RAM or not.

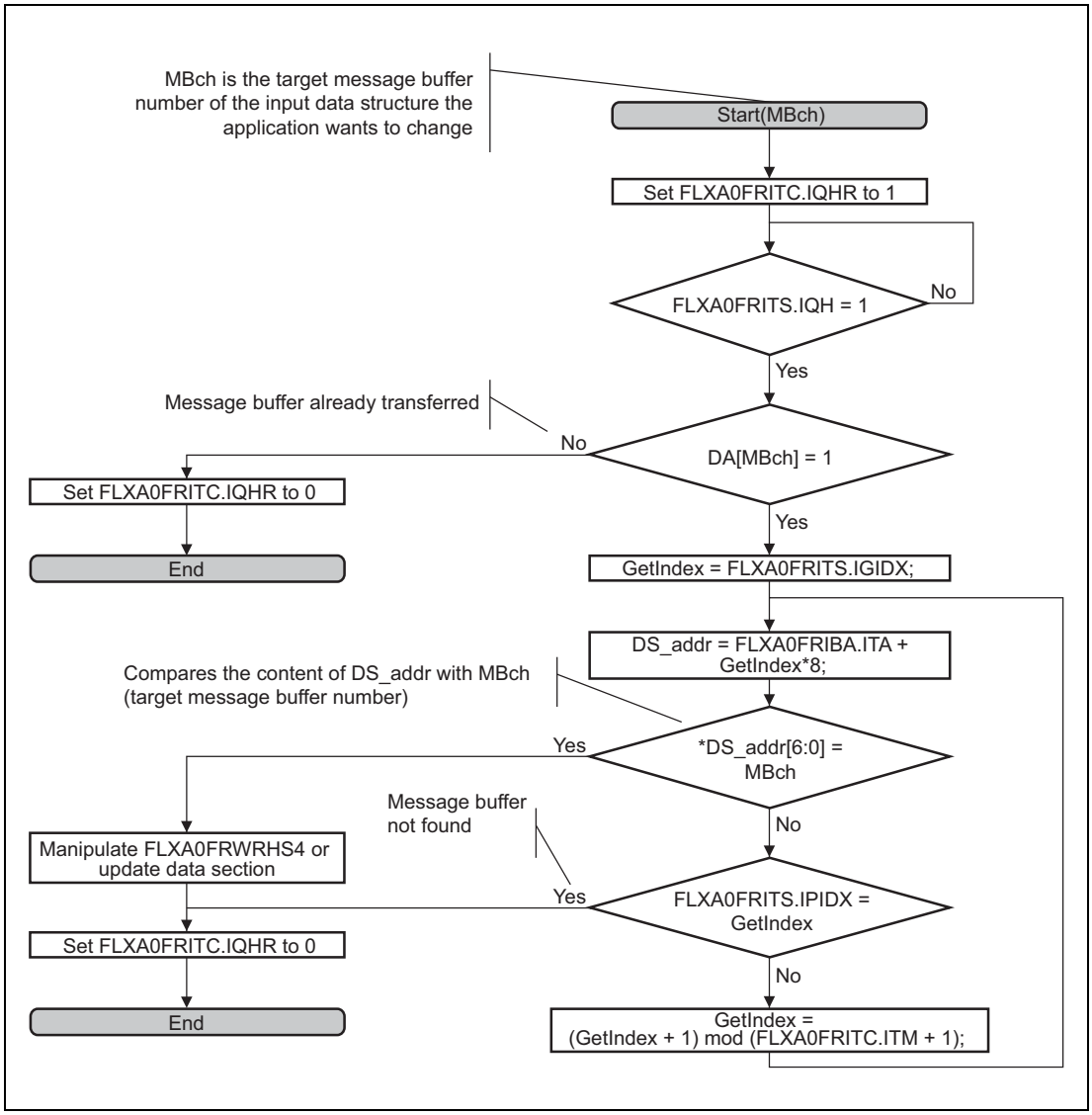


Figure 17.25 Input Table Analysis

In case the message buffer was already transferred to the FlexRay internal message RAM, the user input transfer request can be used to bypass the actual queue and update the required message buffer (see **Section 17.3.16.1 (6), Transfer function of user requested input transfers**).

(6) Transfer function of user requested input transfers

To use this function, the input transfer has to be activated (see **Section 17.3.16.1 (1), Activation and deactivation**).

The application is capable, by using FLXA0FRUIR.UIDX, to request a transfer of an input data structure. The user input transfer request is executed first.

The application has to write the pointer and control field (table entry) to the data structure to be transferred into the input pointer table. Add the table entry for the user input transfer request after the end of the input pointer table (see **Section 17.3.16.1 (3), Input pointer table**).

To transmit this table entry to the input handler, the application has to write the index (FLXA0FRITC.ITM + 1) to the user input transfer request register (FLXA0FRUIR.UIDX).

By writing to the user input transfer request register, the user input transfer request pending flag (FLXA0FRITS.UIRP) is automatically set to ‘1’.

As long this flag is ‘1’, the application cannot make any further user input transfer requests.

The user input transfer request pending flag (FLXA0FRITS.UIRP) changes from ‘1’ to ‘0’ when the requested input transfer is completed. Subsequently, next the pending transfers are processed.

17.3.16.2 Output Data Transfer

When the output data transfer function is enabled, received messages (either in dedicated message buffers or in the FlexRay receive FIFO) are transferred to the local RAM/global RAM by the output data handler. The output data handler can also transfer the message buffer content to the local RAM/ global RAM on application request. When enabled the output handler is also capable to initiate a transfer when the message buffer status has changed.

(1) Activation and deactivation

The output data transfer function should be activated before usage. The activation of the output transfer hander will initialize the FIFO put and get index pointer and FIFO fill level (FLXA0FROTS.FGIDX and FLXA0FROTS.FFL) to zero, set the bits FLXA0FROTS.FDA, FLXA0FROTS.OWP, FLXA0FROTS.FWP and FLXA0FROTS.UORP to ‘0’. Also the interrupt status flags FLXA0FROTS.OTIS, FLXA0FROTS.FIS, FLXA0FROTS.OWIS and FLXA0FROTS.FWIS) are set to ‘0’.

The activation has no effect on the data available flags (FLXA0FRDAi.DA) which are related to the dedicated buffers; these flags have to be cleared by the application.

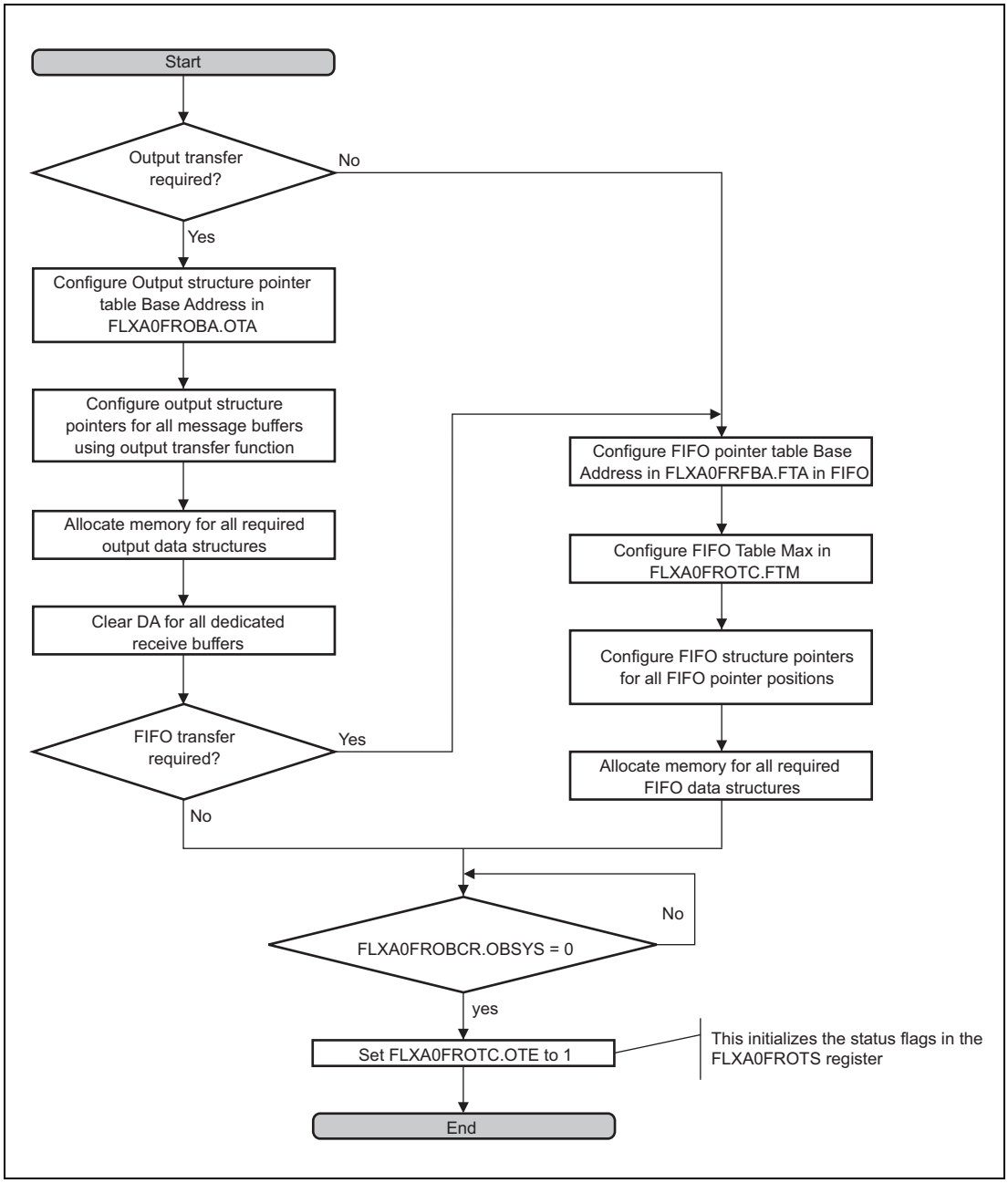


Figure 17.26 Output Transfer Enable Flow

A deactivation request of the output data transfer function can be made at any time. An ongoing transfer will be completed and the completion of this transfer will be flagged. During this time FLXA0FROTS.OTS remains ‘1’.

When FLXA0FROTS.OTS changes from ‘1’ to ‘0’, the output transfer function is deactivated. The data available status flags and the FIFO get index are still maintained when the output transfer function is disabled.

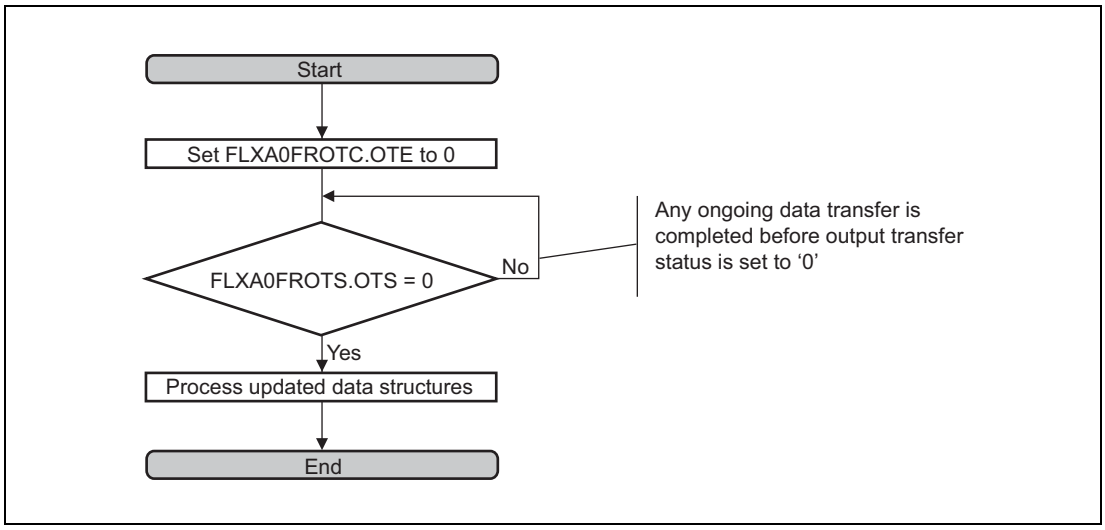


Figure 17.27 Output Transfer Disable Flow

(2) Output transfer data structure

The data in the local RAM/global RAM is stored in an output data structure. The location of the output data structures are determined by output data structure pointers also located in the local RAM/global RAM. The output data structure and indexing is visualized in **Figure 17.28**.

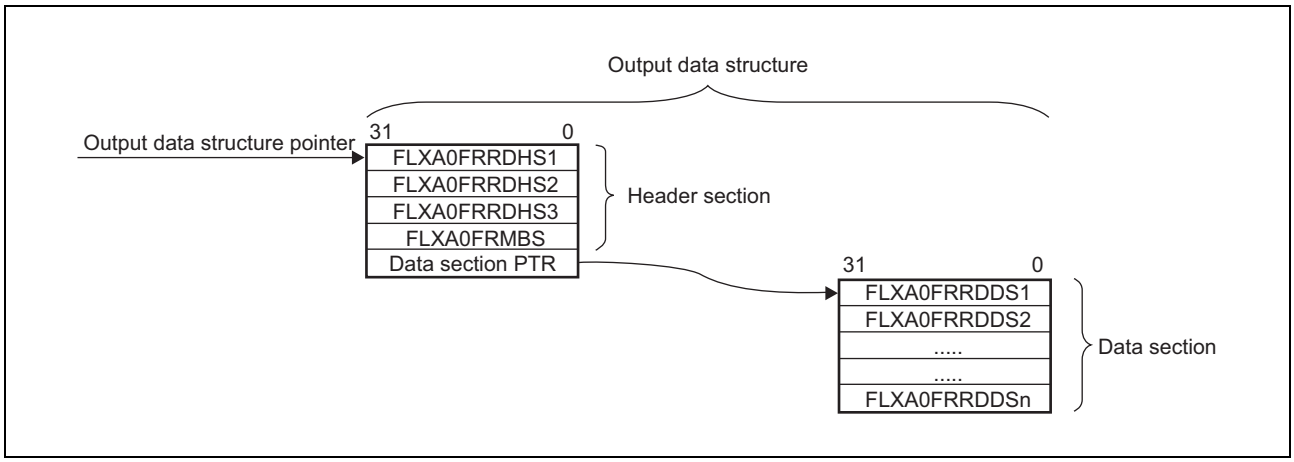


Figure 17.28 Output Data Structure

The output data structure consists of two sections, the header and data section. The header section consists of FLXA0FRRDHS1, FLXA0FRRDHS2, FLXA0FRRDHS3, FLXA0FRMBS and the data section pointer. FLXA0FRRDHS1 is the first element of the structure and has to be aligned to a 32 bit address. The data structure pointer is a reference to the address of FLXA0FRRDHS1. For information about the bit alignment and bit function within the header section, refer to **Section 17.3.13.1, Header Partition**.

FLXA0FRRDDS1 is the first element of the data section. The data section pointer is a reference to the address of FLXA0FRRDDS1 and has to be aligned to a 32 bit address. The byte order for the FlexRay payload data in the output data structure is determined by the bit BEC in the FLXA0FROC register. For information about the payload data alignment within the data section, refer to **Section 17.3.13.2, Data Partition**.

The length of the data section and the total structure size to be allocated in the local RAM/global RAM depend on the configured payload length (FLXA0FRRDHS2.PLC) of the related message buffer. In case the configured payload length is an odd number of words or the received payload length (FLXA0FRRDHS2.PLR) is shorter than the configured payload length, the remaining data words in the local RAM/global RAM are unused and cannot be used by the application.

The output data structure is unique for all three kinds of output transfers. In case only the header section is transferred, the data section pointer is not evaluated by the output handler and the data section remains unchanged.

(3) Output pointer table

For the output data transfer function, the application needs to set up an output pointer table in the local RAM/global RAM. The location of the first element of this table must be programmed into the output pointer table base address (FLXA0FROBA.OTA). This base address has to be aligned to a 32 bit address.

The size of the output pointer table is defined by the maximum of: the last configured dedicated message buffer and the highest message buffer number which will be used for the user output transfer request.

The output pointer table holds pointers (output data structure pointers) to the local RAM/global RAM location where a memory space is reserved for the target message buffer content (header section and data section).

There is a fixed linear relationship between the address of the entries in the output pointer table and the number of the related message buffers (see **Figure 17.29**): the output pointer table starts with the entry for message buffer number 0 at the address configured in FLXA0FROBA.OTA and continues in ascending order for each following message buffer number, by 32 bit aligned address (e.g. message buffer 1 at address OTA + 4, message buffer 2 at address OTA + 8, etc.) for all possible message buffers.

When a set ND bit is the only transfer condition (FLXA0FROTC.OTCS is set to 0), only message buffers configured as a dedicated receive buffer or that will be used for user output transfer requests need have valid pointer entries.

When a set ND bit or a set MBC bit is the transfer condition (FLXA0FROTC.OTCS is set to 1), all dedicated receive buffer and dedicated transmit buffers need to have valid pointer entries.

(4) FIFO output pointer table

The FlexRay module internal FIFO can be extended by a queued buffer structure in the local RAM/ global RAM.

If the FlexRay module internal FIFO is used, the application needs to set up the FIFO output pointer table. The location of the first element of this table is identified by the FIFO pointer table base address (FLXA0FRFBA.FTA). This base address has to be aligned to a 32 bit address.

The size of the FIFO pointer table and hence the maximum number of messages that can be added to the queue, is defined by FIFO Table Max (FLXA0FROTC.FTM).

The FIFO pointer table holds pointers (output data structure pointers) to the local RAM/global RAM location where a memory space is reserved the target message buffer content (header section and data section). For each table entry a data pointer shall be configured in this table.

(5) Transfer function of dedicated message buffers

To use this transfer function, the output transfer has to be activated (see **Section 17.3.16.2 (1), Activation and deactivation**). The activation process requires to set up the output pointer table (see **Section 17.3.16.2 (3), Output pointer table**) in order to specify the destination location (output data structures) for the data to transfer. **Figure 17.29** shows how the output pointer table references the output data structures.

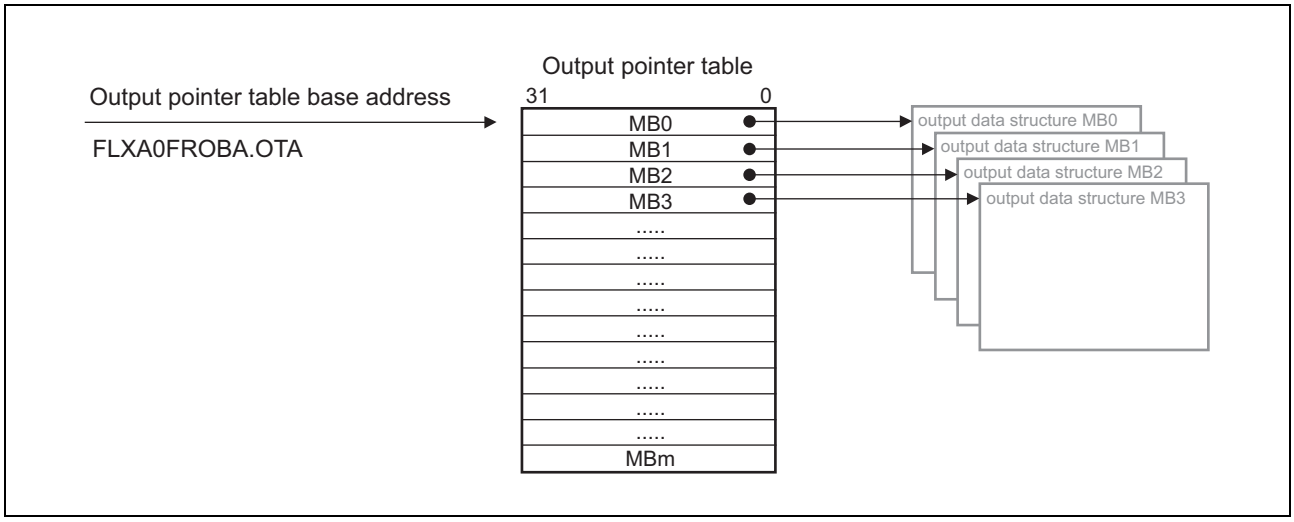


Figure 17.29 Output Data Structure and Indexing

With FLXA0FROTC.OTCS the output transfer condition can be selected between the ‘New data only mode’ and the ‘New data and status changed mode’.

In the ‘New data only mode’, an output data transfer is initiated when a valid FlexRay data frame has been stored into a dedicated receive buffer which causes the related ND flag in the FLXA0FRNDATi register to set. The ND flag in the FLXA0FRNDATi register is automatically set to ‘0’ during the transfer procedure. The header section is also transferred and hence the MBC flag in the FLXA0FRMBSCi register is set to ‘0’.

In the ‘New data and status changed mode’, an output data transfer is initiated as described in the ‘New data only mode’. In addition an output data transfer is initiated when only the message buffer status has been changed which causes the related MBC flag in the FLXA0FRMBSCi register to be set. In this case only the header section is transferred. The flag in the register is automatically set to ‘0’ during the transfer procedure.

After transferring the message buffer data from the FlexRay internal message RAM to the output data structure, the corresponding data available flag in the FLXA0FRDAi (i = 0 to 3) registers is set to ‘1’. The update of the output data structure is also flagged by the setting of the output transfer interrupt status flag (FLXA0FROTS.OTIS).

As long as the data available flag remains ‘1’, the corresponding output data structure will not be updated.

In the case

- the data available flag is ‘1’ and a valid received message was stored or
- when FLXA0FROTC.OTCS is ‘1’ and the message buffer status was updated,

The output transfer warning interrupt flag (FLXA0FROTS.OWIS) is set to ‘1’ notifying the application that new data is available but the output data structure transfer cannot be processed. In addition FLXA0FROTS.OWP is set to ‘1’ that continuously flags that status of the output transfer warning condition.

If a valid receive message in the FlexRay internal message RAM is overwritten by an additional receive message, the message lost flag (FLXA0FRMBS.MLST) is set to ‘1’. This flag can be evaluated after the message buffer has been transferred into an output data structure.

The following sections describe how output data structures can be handled.

(a) Data section copy method

One option is to copy the information from the output data structure to a different location of the local RAM/global RAM and then to release the output data structure by clearing the related data enable flag. The application can use the copied information for further processing.

The C-code below is an example showing how output data structures could be handled in SW.

```
// Get copy of one output structure block (FlexRay Rx or user transfer)
// Return -1 if the buffer has no valid data (DA = "0") else the used
payload size
int flex_copy_output_block(unsigned mbnr, void *target_hs, void
*target_ds)
{
    unsigned long *this_FRDA, mask;
    int size;

    //ckeck if data of MB is available
    this_FRDA = (&FRDA0) + ((mbnr >> 5) & 0x03);
    mask = 1 << (mbnr & 0x1f)
    if ((*this_FRDA & mask) == 0) return -1;

    //calculate the real MB size min(PLC, PLR)
    size = flex_output_table[mbnr]->hs2.f.plr;
    if (size > flex_output_table[mbnr]->hs2.f.plc)
        size = flex_output_table[mbnr]->hs2.f.plc;

    //copy header and data to target location
    if (target_hs != NULL)
        memcpy(flex_output_table[mbnr], target_hs, 4*4);
    if (target_ds != NULL)
        memcpy(flex_output_table[mbnr]->dsptr, target_ds, 2*size);

    //acknowledge reception
    *this_FRDA = mask;      /* clear by writing a "1" */

    //done
    return size;
}
```

(b) Data structure pointer method

A different option is to modify the output data structure pointer in the output pointer table and to release the output data structure by clearing the related data enable flag. The changed output data pointer should refer to a free data structure. The application needs to use the old data structure for further processing.

The C-code below is an example showing how output data structures could be handled in SW.

```
// Get one output structure block (FlexRay Rx or user transfer)
// The returned block is replaced by a free output structure
// Return the block or NULL if the buffer has no valid data (DA = "0")
t_os_block *flex_get_output_block(unsigned mbnr, t_os_block *free_blk)
{
    t_os_block *this_blk;
    unsigned long *this_FRDA, mask;

    //ckeck if data of MB is available
    this_FRDA = (&FRDA1) + ((mbnr >> 5) & 0x03);
    mask = 1 << (mbnr & 0x1f);
    if ((*this_FRDA & mask) == 0) return NULL;

    //get pointer from output table and replace it by free location
    this_blk = flex_output_table[mbnr];
    flex_output_table[mbnr] = free_blk;

    //acknowledge reception
    *this_FRDA = mask;      /* clear by writing a "1" */

    //done
    return this_blk;
}
```

(c) Data section pointer method

A third option is to modify the data section pointer in the output data structure and to release the output data structure by clearing the related data enable flag. The changed data section pointer should refer to a free memory area. The application should use the old data section for further processing by forwarding the data section pointer.

The C-code below is an example showing how output data structures could be handled in SW.

```
// Get one data section block (FlexRay Rx or user transfer)
// The returned data section block is replaced by a free data section
area
// Return the block or NULL if the buffer has no valid data  (DA = "0")
t_ds_block *flex_get_datasection_block(unsigned mbnr, t_ds_block
*free_blk)
{
    t_os_block *this_blk;
    unsigned long *this_FRDA, mask;
    unsigned long this_os_ptr;

    //ckeck if data of MB is available
    this_FRDA = (&FRDA1) + ((mbnr >> 5) & 0x03);
    mask = 1 << (mbnr & 0x1f);
    if ((*this_FRDA & mask) == 0) return NULL;

    //get base address from output table
    this_os_ptr = (unsigned long)flex_output_table[mbnr];

    //get pointer from output data section and replace it by free
location
    this_blk = (t_os_block*)(this_os_ptr + 20);
    (t_os_block*)(this_os_ptr + 20) = free_blk;

    //acknowledge reception
    *this_FRDA = mask;      /* clear by writing a "1" */

    //done
    return this_blk;
}
```

(6) Transfer function of FIFO message buffers

To use this buffer transfer function, the output transfer has to be activated (see **Section 17.3.16.2 (1), Activation and deactivation**). The activation process requires the setup of the FIFO pointer table (see **Section 17.3.16.2 (4), FIFO output pointer table**) in order to specify a location in the local RAM/global RAM reserved for the storage of the required output data structures.

A FIFO data transfer is initiated when a valid FlexRay data frame has been stored in the FlexRay internal FIFO.

After transfers from the internal FIFO to the output data structure, the FIFO interrupt status flag (FLXA0FROTS.FIS) and FLXA0FROTS.FDA are set to '1'. The bit FLXA0FROTS.FIS can be used as an interrupt source. The bit FLXA0FROTS.FDA indicates that the FIFO is not empty.

Up to FLXA0FROTS.FTM output structures configured in the register can be queued.

The transfer to the extended FIFO buffer structure is controlled by index pointers. This put index is controlled by the FIFO transfer handler and is incremented after transferring a message to the output data structure. The FIFO reception handler also maintains a get index which is flagged in FLXA0FROTS.FGIDX. The value of this get index is known by the application by either reading the status or maintaining a software variable. The get index (initially set to the zero) is incremented by one when the application releases the oldest entry of the FIFO queue by writing '1' to FLXA0FROTS.FDA. By comparing the put index and the get index, the FIFO handler acquires the current fill level of the queued buffer structure.

The current FIFO fill level is set in FLXA0FROTS.FFL. When FLXA0FROTS.FDA is '1', there is at least one entry in the FIFO queue.

In case the queued buffer structure in the local RAM/global RAM is full (FLXA0FROTS.FFL = FLXA0FROTC.FTM + 1), no further transfers are initialized, new messages remain in the FlexRay internal FIFO and the FIFO transfer warning interrupt status flag (FLXA0FROTS.FWIS) is set to '1'.

In case the FlexRay internal FIFO structure becomes full, messages in the FlexRay internal FIFO structure will be overwritten. The related status flags and configuration registers of the FlexRay core module are used to generate desired warning notifications.

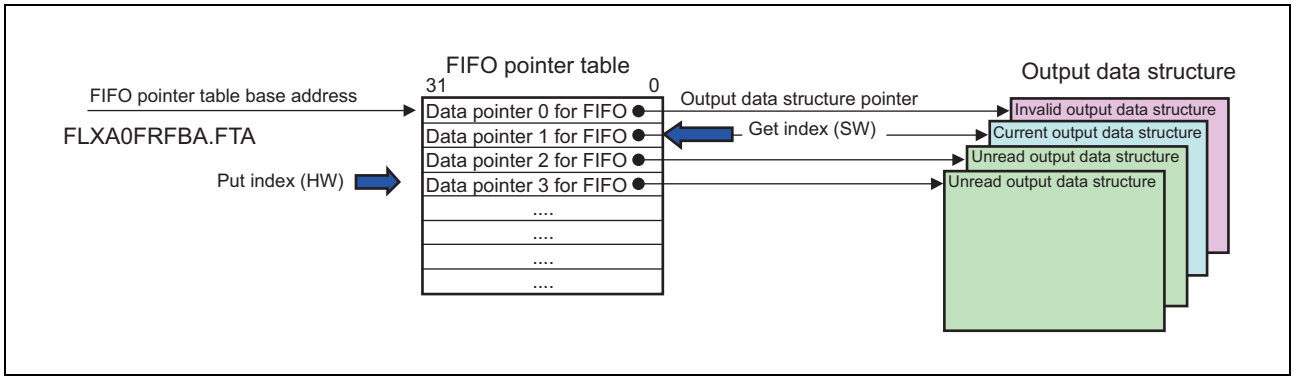


Figure 17.30 FIFO Pointer Table

The following code illustrates a good example of handling the extended FIFO. The function flex_get_fifo_block ( ) assumes that there is a global variable flex_fifo_gidx which is initialized to zero when the output transfer is enabled. Using this variable, a read access to FLXA0FROTS.FGIDX becomes redundant. The constant FLEX_FIFO_MAX is used to configure FLXA0FROTC.FTM. When getting an output structure from the FIFO queue for further processing, its pointer is replaced by an empty output data structure given as parameter.

```
// Get one output structure block from the FlexRay Rx FIFO
// The returned block is replaced by a free output structure
// Return the block or NULL if FIFO is empty
t_os_block *flex_get_fifo_block(t_os_block *free_blk)
{
    t_os_block_ptr this_blk;

    //ckeck if data in FIFO is available
    if (FROTS.FDA == 0) return NULL;

    //get pointer from FIFO table and replace it by free location
    this_blk = flex_fifo_table[flex_fifo_gidx];
    flex_fifo_table[flex_fifo_gidx] = free_blk;

    //increment SW get index
    if (++flex_fifo_gidx == FLEX_FIFO_MAX) flex_fifo_gidx = 0;

    //acknowledge FIFO reception
    FROTS.FDA = 1;      /* clear by writing a "1" */

    //done
    return this_blk;
}
```

(7) Transfer function of user output transfer requests

To use this transfer function, the output transfer has to be activated (see **Section 17.3.16.2 (1), Activation and deactivation**). The activation process requires to set up the output pointer table (see **Section 17.3.16.2 (3), Output pointer table**) in order to specify the location in the local RAM/ global RAM reserved for the transfer of the data (output data structures).

The application can, by using FLXA0FRUOR.UMBNR, request a transfer of dedicated message buffer to an output data structure. Except in CONFIG state, message buffers which are part of the FlexRay internal FIFO cannot be requested.

The header section is always transferred to the output data structure. The transfer of the data section can be enabled by setting FLXA0FRUOR.URDS to '1'. The selected message buffer content is stored in the output data structure location determined by the pointers in the output pointer table.

The data available status and transfer blocking by the DA bits in the FLXA0FRDAi register is also used for the user requested transfers. Therefore, the DA bit in the FLXA0FRDAi register related to the requested buffer number (FLXA0FRUOR.UMBNR) needs to be released before making the transfer request.

After writing to FLXA0FRUOR.UMBNR, the bit FLXA0FROTS.UORP is set to '1' to indicate that there is a pending user transfer request. When the transfer has been processed the bit FLXA0FROTS.UORP is set to '0', the bit FLXA0FROTS.OTIS is set to '1' and the DA bit in the FLXA0FRDAi register related to the requested buffer number (FLXA0FRUOR.UMBNR) is set to '1'.

User output transfer requests cannot be queued. The application needs to check the bit FLXA0FROTS.UORP before writing to FLXA0FRUOR.UMBNR.

User output transfer requests cannot be made for message buffers which are pending in the input transfer queue.

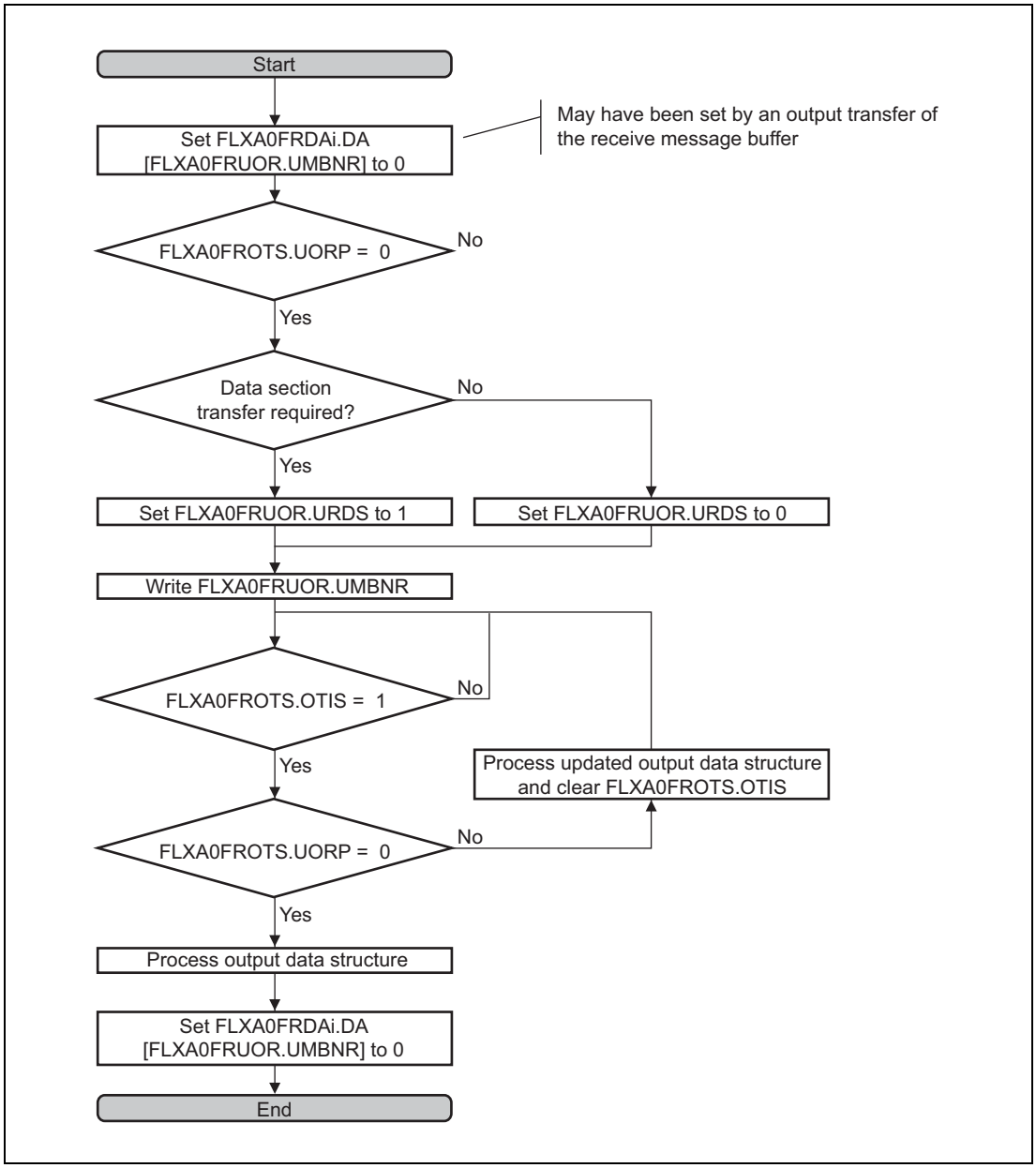


Figure 17.31 User Output Transfer Request Flow

Note that it may be possible that the data structure addressed by a user request is being updated due to a receive message buffer update (which causes the bit DA in the FLXA0FRDAi register being set). This DA flag setting inhibits the user output transfer request. Therefore, polling FLXA0FROTS.UORP is not a secure method to identify when the transfer of a requested message buffer has been completed. The bits FLXA0FROTS.OTIS or the DA bit in the FLXA0FRDAi register can be used instead. The exact flow depends on the software architecture.

17.3.16.3 Data Structure Transfer Scheduling

Cyclically the different types of transfer requests are checked. In order to guarantee a certain transfer time, the different types of transfers have different priorities.

Use requested input transfers that have highest priority followed by the transfer of data structures to be transmitted into the active input transfer queue. No new output transfer will be started as long as there is a pending input transfer request.

The three output transfer request types are checked in a specific order:

(1) All dedicated message buffers in ascending order

When FLXA0FROTC.OTCS is set to ‘0’, set flags in the FLXA0FRNDATn register are causing a transfer of the message buffer to the output data structure if the destination area is free (DA bit in the FLXA0FRDAi register is ‘0’).

When FLXA0FROTC.OTCS is set to ‘1’, set flags in the FLXA0FRNDATn register or set flags in the FLXA0FRMBSCn register are causing a transfer of the message buffer to the output data structure if the destination area is free (DA bit in the FLXA0FRDAi register is ‘0’).

(2) FlexRay internal FIFO

When the FlexRay internal FIFO is not empty and there is a free destination area, one FIFO message is transferred into the output data structure addressed by the FIFO pointer table.

(3) User output request

If there is a pending user output transfer request, one message buffer is transferred into the corresponding output data structure.

The check sequence is suspended when an input transfer is generated.

17.3.16.4 Behavior in Case of Data Transfer Access Error

The memory areas accessed by the data transfer function may be protected by a memory protection unit (MPU). When the MPU flags an access to a protected address caused by an input or output transfer, an access error event is generated and the related bit in the FLXA0FRAES register is set.

The ongoing transfer is immediately terminated but completed transfers are processed and may generate further access errors. Any following access errors are only flagged in FLXA0FRAES.MAE. The other status flags are not updated.

(1) Access error during input transfer

When an access error occurs during an input transfer:

- The ongoing transfer is immediately terminated. The FlexRay internal message RAM will not be updated
- The address, the FlexRay module intended to access to, is written in the FLXA0FRAEA register
- FLXA0FRAES.IAE is set to '1'
- The input pointer table index is flagged in FLXA0FRAES.EIDX
- In case of a normal input transfer the transfer related DA bit in the FLXA0FRDAi register is set to '0'
- In case of a user input transfer request FLXA0FRITS.UIRP is set to '0'

With the given status information, the application is able to identify and correct the faulty data structure.

In addition the application needs to clear the input access error flag (FLXA0FRAES.IAE).

(2) Access error during output transfer

When an access error occurs during an output transfer:

- The ongoing transfer is immediately terminated but the update of the data structure may be started.
- The address, the FlexRay module intended to access to, is written in the FLXA0FRAEA register
- FLXA0FRAES.OAE is set to '1'
- The output pointer table index is flagged in FLXA0FRAES.EIDX
- In case of a normal output transfer, the related DA flag in the FLXA0FRDAi register remains '0' and no output transfer interrupt is generated
- In case of a user output transfer request, FLXA0FROTS.UORP is set to '0'

With the given status information, the application is able to identify and correct the faulty data structure. The data structure in the local RAM/global RAM cannot be treated as valid.

In addition the application needs to clear the output access error flag (FLXA0FRAES.OAE).

The FlexRay module internal transfer of the message buffer is completed before the local RAM/global RAM access error is detected. The output transfer cannot be resumed. To avoid loss of data, the application can perform a user output transfer request of this message buffer to a correct local RAM/ global RAM location.

(3) Access error during FIFO transfer

When an access error occurs during an FIFO transfer:

- The ongoing transfer is immediately terminated
- The address, the FlexRay module intended to access to, is written in the FLXA0FRAEA register
- FLXA0FRAES.FAE is set to '1'
- The FIFO pointer table index is written in FLXA0FRAES.EIDX
- The FIFO index pointer is not changed and hence the FIFO status flags are unchanged

With the given status information, the application is able to identify and correct the faulty data structure.

In addition, the application needs to clear the FIFO access error flag (FLXA0FRAES.FAE).

The data in the local RAM/global RAM cannot be treated as valid and is not released to the application. The message cannot be recovered.

17.3.16.5 Behaviors in Case of RAM Read Errors

The FlexRay internal message RAM has an ECC checking mechanism. In case an uncorrectable RAM read error occurs, the application has to analyze the status in the FLXA0FRMHDS register and redo as described in **Section 17.3.16.3, Data Structure Transfer Scheduling**. The input and output transfer handler redoes also on these errors detected in the message RAM when the error is related to an active transfer.

In addition, temporary buffer RAM A and temporary buffer RAM B have an ECC checking mechanism as well.

An uncorrectable RAM read errors does not affect the data transfer functionality but have to be handled as described in **Section 17.3.13.1 (4), Message Buffer Status FLXA0FRMBS (word 3)**.

In all cases, data causing a read error is never transferred to the local RAM/global RAM. If there is no recovery available in the application, the message is lost.

(1) Read error during transfer from TBF to MBF

This internal transfer is performed for each valid FlexRay message received.

A read error can only occur when the header section in the FlexRay message RAM (see read error flags in FLXA0FRMHDS) is read. In this case, the message buffer needs to be re-configured.

For dedicated receive message buffers, the related ND flag in the FLXA0FRNDATi register will not be set. Consequently the affected message buffer will not be transferred to the output data structure.

For the FlexRay internal FIFO buffers, the ND flag in the FLXA0FRNDATi register is not set, but the FlexRay internal FIFO put index is incremented. Due to this, a transfer procedure from the FlexRay internal FIFO buffer to the output buffer is started. However, if the read error is still present in the header section, updating of the output data structure will not start (see **Section 17.3.16.5 (2), Read error during transfer from MBF to OBF**); thus the data in the local RAM/global RAM remains correct.

Note that the correction or any other reconfiguration of FIFO related to message buffers, while there are pending FIFO transfers, may result in incorrect data in the local RAM/global RAM. It is strongly recommended to deactivate the output data transfer before starting the reconfiguration and flush the FlexRay internal FIFO before reactivation of the output data transfer.

(2) Read error during transfer from MBF to OBF

This internal transfer is performed for every output data transfer (dedicated reception, FIFO, user requested).

A read error can occur in the header and data section (see read error flags in FLXA0FRMHDS). In both cases, the message gets lost. If the error is located in the header section, the message buffer needs to be re-configured. If the error is located in the data section, the error is corrected with the next data section update.

When a read error occurs during the transfer from the message RAM to the output buffer, the output data structure will not be updated and the data enable bit will not be set to '1'. The FIFO put index and the FIFO fill level are not changed also. In case of user output transfer request, FLXA0FROTS.UORP is set to '0' even if there was no update of the output data structure.

(3) Read error during transfer from IBF to MBF

This internal transfer is performed for every input data transfer.

A read error can occur only when the header section requested (the bit LHS in FLXA0FRWRHS4 is set to "0") cannot be updated due to the reading of the header section from the message RAM (see read error flags in FLXA0FRMHDS). In this case, the message buffer needs to be re-configured.

When a read error occurs during the input data transfer, the actually transferred message in the input queue is lost.

(4) Message RAM read errors

Read errors when reading the header section are flagged in the FLXA0FRMHDS register.

Depending on the buffer type and set buffer protection, the message buffer may not be reconfigured.

The input transfer function cannot be used to reconfigure a locked message buffer using the method described in **Section 17.3.13.4 (3), Temporary Unlocking of Header Section**.

Before reconfiguring a locked buffer, the user needs to disable the input transfer function and the output transfer function.

17.3.16.6 Data Transfer Timings

The transfer timing between the local RAM/global RAM and the FlexRay internal message buffers is composed of two elements. The first element is the FlexRay core internal transfer time between the interface buffer and the message RAM. The second element is the transfer time required between the local RAM/global RAM and the interface buffer.

The concept of the input- and output handler allows only one active transfer between the local RAM/ global RAM and input buffer as well as only one active transfer between the input buffer and the message RAM. In case two requests are made at the same time, the input transfer task is handled at first.

Due to this concept constrains the calculation of transfer times does not consider a parallel transfer but the worst case scenario whereby an output transfer is started only after all input transfers are completed.

(1) FlexRay core internal transfer time

The number of clock cycles required for a transfer is set in FlexRay core module clock cycles (bus clock). The transfer time differs between the input and output transfer.

The time required for a transfer between the interface buffer and the FlexRay internal message RAM is given by [ERAY_ADD]

$$cycles_{req\_MHD[PL]} = (numberOfConcurrentTasks) * (setupTime + (DataWords_{req[PL]}) + 75)$$

Equation 3

The number of data words (32 bit) can be calculated by

$$DataWords_{req[PL]} = 4 + floor\left(\frac{PLC + 1}{2}\right)$$

Equation 4

Whereby 4 is the number of header section words (32 bit) which needs to be always accessed even if the header section is not updated.

As there can be in total three parallel tasks in the FlexRay internal message handler but the number of concurrent tasks is unknown, not only for a worst case consideration the number of concurrent tasks must be set to 3. Beside the setup time and the number of words to be transferred a message handler internal delay time of 75 bus clock cycles must be added. The maximum number of payload to be transferred is in worst case 64.

The worst case for an input transfer is different from the worst case for an output transfer.

For an input transfer the setup time required is 6 bus clock cycles and therefore

$$cycles_{req\_MHD\_input[64]} = 3 * (6 + 68 + 75) = 447$$

Equation 5 Message hander internal input transfer time (longest case)

For an output transfer the setup time required is 2 bus clock cycles and therefore

$$cycles_{req\_MHD\_output[64]} = 3 * (2 + 68 + 75) = 435$$

Equation 6 message handler internal output transfer time (longest case)

(2) System bus transfer time

The number of bus clock cycles required for a transfer is written in bus clock cycles and is identical for all kind of transfers between the local RAM/global RAM and the FlexRay module.

The time required by the transfer handler for a transfer between the local RAM/global RAM and the FlexRay interface buffer is given by

$$cycles_{req\_SB[PL]} = throughput * (systemTime + DataWords_{req[PL]}) + setupTime$$

Equation 7 General system bus transfer time formula

The number of payload words (32 bit) can be calculated by using Equation 4.

The throughput depends on the utilization of the system bus and depends on the application. However, the data transfer handler has the highest priority for accessing the local RAM/global RAM and the data throughput factor can be assumed as XXXX even in worst case scenarios. The system time (read of the data pointer from local RAM/global RAM) can be set to 1 in all cases. The transfer setup time can be set to 3 in all cases.

Therefore, the system bus transfer time (given in bus clock cycles) is given by

$$cycles_{req\_SB[64]} = throughput * (2 + 68) + 3 = XXXX * (2 + 68) + 3$$

Equation 8 System bus transfer time (longest case)

In addition to the system bus transfer time, a FlexRay module internal processing time needs to be taken into account. The number of clock cycles required for the internal processing time required is given in FlexRay core module clock cycles (bus clock) and is different between input transfers and output transfers but can be treated as constants.

$$cycles_{req\_THD\_input} = 2 + 4 = 6$$

Equation 9

Whereby 2 cycles are required to initiate the FlexRay module internal transfer from the input buffer to the message RAM. In addition 4 internal processing cycles are required.

$$cycles_{req\_THD\_output} = 3 + 4 = 7$$

Equation 10

Whereby 3 cycles are required to initiate the FlexRay module internal transfer from the input buffer to the message RAM. In addition 4 internal processing cycles are required.

(3) Summary required input transfer time

The worst case time (as a function of the transferred payload) required to transfer data between the local RAM/global RAM to the FlexRay internal message buffer can be calculated by

$$t\_input_{[PL]} = t\_chi * (cycles_{req\_THD\_input} + cycles_{req\_MHD\_input[PL]}) + t\_sys * (cycles_{req\_SB[PL]})$$

$$t\_input_{[PL]} = t\_chi * \left(6 + 3 * \left[81 + DataWords_{[PL]}\right]\right) + t\_sys * \left(throughput * \left[2 + DataWords_{[PL]}\right] + 3\right)$$

Equation 11 input transfer time

With

$$t\_sys = \left(f(sysclk)\right)^{-1}$$

and

$$t\_chi = \left(f(clkp2\_flx)\right)^{-1}$$

(4) Summary required output transfer time

The worst case time (as a function of the transferred payload) required to transfer data from the FlexRay internal message buffer to the local RAM/global RAM can be calculated by

$$t\_output_{[PL]} = t\_chi * (cycles_{req\_THD\_output} + cycles_{req\_MHD\_output[PL]}) + t\_sys * (cycles_{req\_SB[PL]})$$
$$t\_output_{[PL]} = t\_chi * \left(7 + 3 * \left\lceil 77 + DataWords_{[PL]} \right\rceil\right) + t\_sys * \left(throughput * \left\lceil 2 + DataWords_{[PL]} \right\rceil + 3\right)$$

Equation 12 Output transfer time

With

$$t\_sys = \left(f(sysclk)\right)^{-1}$$

and

$$t\_chi = \left(f(clkp2\_flx)\right)^{-1}$$

17.3.17 Byte Alignment

The alignment of the bytes received by the FlexRay protocol and the alignment of the bytes required by the application may be different. The FlexRay module provides with FLXA0FROC.BEC a byte alignment function to support different byte ordering styles.

Figure 17.32 shows the payload byte alignment in a FlexRay frame.

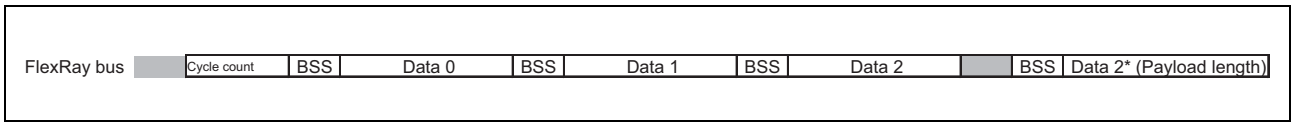


Figure 17.32 Byte Alignment on the FlexRay Bus

17.3.17.1 Little Endian Alignment

When FLXA0FROC.BEC is '0', the byte alignment is set to Little Endian.

(1) FLXA0FRNMVn (n = 1 to 3)

The byte alignment of the NMV bytes is as below.

Bit	31 30 29 28	27 26 25 24	23 22 21 20	19 18 17 16	15 14 13 12	11 10 9 8	7 6 5 4	3 2 1 0
FLXA0FRNMV1	Data 3		Data 2		Data 1		Data 0	
FLXA0FRNMV2	Data 7		Data 6		Data 5		Data 4	
FLXA0FRNMV3	Data 11		Data 10		Data 9		Data 8	

(2) FLXA0FRWRDSn (n = 1 to 64)

The byte alignment for the message payload in the FlexRay input buffer and the input data structure is as below.

FLXA0FRWRDSn.MD[7:0] = $Data_{4n-4}$

FLXA0FRWRDSn.MD[15:8] = $Data_{4n-3}$

FLXA0FRWRDSn.MD[23:16] =$Data_{4n-2}$

FLXA0FRWRDSn.MD[31:24] = $Data_{4n-1}$

Transmission order on the FlexRay bus is FLXA0FRWRDSn.MD[7:0], FLXA0FRWRDSn.MD [15:8], FLXA0FRWRDSn.MD [23:16], FLXA0FRWRDSn.MD [31:24] with the most significant bit (MSB) transmitted first.

(3) FLXA0FRRDDSn (n = 1 to 64)

The byte alignment for the message payload in the FlexRay output buffer and the output data structure is as below.

FLXA0FRRDDSn.MD[7:0] = $Data_{4n-4}$

FLXA0FRRDDSn.MD[15:8] = $Data_{4n-3}$

FLXA0FRRDDSn.MD[23:16] =$Data_{4n-2}$

FLXA0FRRDDSn.MD[31:24] = $Data_{4n-1}$

Transmission order on the FlexRay bus is FLXA0FRRDDSn.MD[7:0], FLXA0FRRDDSn.MD[15:8], FLXA0FRRDDSn.MD[23:16], FLXA0FRRDDSn.MD[31:24] with the most significant bit (MSB) transmitted first.

17.3.17.2 Big Endian Alignment

When FLXA0FROC.BEC is '1', the byte alignment is set to Big Endian.

(1) FLXA0FRNMVn (n = 1 to 3)

The byte alignment of the NMV bytes is as below.

Bit	31 30 29 28	27 26 25 24	23 22 21 20	19 18 17 16	15 14 13 12	11 10 9 8	7 6 5 4	3 2 1 0
FLXAnFRNMV1	Data 0		Data 1		Data 2		Data 3	
FLXAnFRNMV2	Data 4		Data 5		Data 6		Data 7	
FLXAnFRNMV3	Data 8		Data 9		Data 10		Data 11	

(2) FLXA0FRWRDSn (n = 1 to 64)

The byte alignment for the message payload in the FlexRay input buffer and the input data structure is as below.

FLXA0FRWRDSn.MD[7:0] = $Data_{4n-1}$

FLXA0FRWRDSn.MD[15:8] = $Data_{4n-2}$

FLXA0FRWRDSn.MD[23:16] =$Data_{4n-3}$

FLXA0FRWRDSn.MD[31:24] = $Data_{4n-4}$

Transmission order on the FlexRay bus is FLXA0FRWRDSn.MD[31:24], FLXA0FRWRDSn.MD[23:16], FLXA0FRWRDSn.MD[15:8], and FLXA0FRWRDSn.MD[7:0] with the most significant bit (MSB) transmitted first.

(3) FLXA0FRWRDSn (n = 1 to 64)

The byte alignment for the message payload in the FlexRay output buffer and the output data structure is as below.

FLXA0FRRDDSn.MD[7:0] = $Data_{4n-1}$

FLXA0FRRDDSn.MD[15:8] = $Data_{4n-2}$

FLXA0FRRDDSn.MD[23:16] =$Data_{4n-3}$

FLXA0FRRDDSn.MD[31:24] = $Data_{4n-4}$

Transmission order on the FlexRay bus is FLXA0FRWRDSn.MD[7:0], FLXA0FRWRDSn.MD [15:8], FLXA0FRWRDSn.MD [23:16], FLXA0FRWRDSn.MD [31:24] with the most significant bit (MSB) transmitted first.

Section 18 Renesas High-Speed Bus (RHSB)

18.1 Overview

18.1.1 Features of RHSB

Channels: This microcontroller has the following number of channels of the Renesas high-speed bus (RHSB).

Table 18.1 Channels of RHSB

Renesas High-Speed Bus	RH850/E1M-S
Instance	2
Name	RHSB0, RHSB1

Chip select index: The Renesas high-speed bus has up to 2 chip select signals. Throughout this section, the individual chip select signals are identified by the index “x”, thus a given chip select signal is denoted as RHSBjCSDx (j = 0, 1).

The number of chip select signals for each channel of RHSBj is given in the following table.

Table 18.2 RHSB Chip Select Index

RHSB instance	RH850/E1M-S
RHSB0	RHSB0CSDx (x = 0, 1)
RHSB1	RHSB1CSDx (x = 0, 1)

Upstream serial input index: The Renesas high-speed bus has up to 2 upstream serial input signals. Throughout this section, the individual upstream serial input signals are identified by the index “x”, thus a upstream serial input signal is denoted as RHSBjSIx (j = 0, 1).

The number of upstream serial input signals for each channel of RHSBj is given in the following table.

Table 18.3 RHSB Upstream Serial Input Index

RHSB Channels	RH850/E1M-S
RHSB0	RHSB0SIx (x = 0, 1)
RHSB1	RHSB1SIx (x = 0, 1)

18.1.2 Module Overview

This section describes the features of this Renesas high-speed bus (RHSB). The RHSB is designed to be used as a master node for up to 4 Microsecond Bus Channels. It is designed to be compliant to MSC 1.0 (BOSCH/Infineon, July 2003) specification and to MSC 1.0.0 (IPextreme, July 27, 2007).

18.1.2.1 Downstream Channel Communication Functions

Downstream communication

- Synchronous serial transmission
- Clock phase is selectable and clock activity is configurable during passive phases
- Configurable repetition period is independent from data frame length (1 to 512_D bits (data bit))
- Configurable downstream bit rate to be $f_{DW} = f_{PE}/x$ with x = (2, 4, 5, 8, 16, 32, 64, 128, 256). f_{PE} is typically 80 MHz.
- Indication of the status of ongoing downstream transmission, for evaluation purpose

Up to 2 slaves are supported with this individual configuration

- Active level of the chip select and data bits
- Selection bit for data frame present/not present
- Length of assertion time (0 to 7_D bits) and of de-assertion time (0 to 7_D bits)

Downstream data frame

- Each data frame (1 to 64 bits) is composed by 1 to 4 DFTEs with individual length
- The user can define for each bit of a DFTE an individual source (e.g. timer input, data register)
- The frame passive phase after each data frame is configurable from 1 to 64 bits
- Three transmission modes to schedule data transmission

Downstream command frame

- Command frame transmission with/without request of remote data
- Dedicated register is available defining 1 to 32 bits for command frame transmission
- Three different methods to insert command frames into the data frame schedule

Emergency function

- Handling of emergency condition can be enabled/disabled
- Emergency condition is identified by an input pin of the MCU with programmable active level
- Emergency input is selectable between level or edge sensitive
- After transmission of an emergency frame data transmission can be stopped automatically
- Emergency data transmission DFTE bits are individually programmable through a dedicated register

18.1.2.2 Upstream Channel Communication Functions

Upstream communication

- Asynchronous serial reception
- Automatic selection of slave configuration when applying a command frame with data request
- Timeout detection can be activated to monitor slave response on a command frame with data request
- Status indication of ongoing upstream reception transmission, for evaluation purpose

Up to 2 slaves are supported for upstream communication with individual configuration

- Active level of data line
- Frame with 8 or 12 data bits, even or odd parity bit, and 2 or 3 stop bits
- Upstream bit rate derived from downstream bit rate ($f_{UP} = f_{DW}/2^x$ with x = 3 to 9; results in /8 to /512)

Upstream status of up to 2 slaves

- Valid data received
- Data lost (overrun of data register)
- Timeout error
- Receive error (parity bit error or stop bit error)

18.1.2.3 Interrupts

The RHSB module has 9 sources for interrupt requests. 5 interrupts are downstream related, 4 interrupts are upstream related. These requests are grouped to 5 physical interrupt request lines.

There is an individual interrupt enable bit and an individual interrupt status bit for each interrupt source.

18.1.2.4 DMA Support

The RHSB module supports an update of the downstream data registers by downstream data DMA transfer. The ‘Data frame transmission started’ interrupt is used to trigger this transfer.

The RHSB module supports an update of the downstream command data registers by downstream command DMA transfer. Command frame transmission done' interrupt is used to trigger this transfer.

The RHSB module supports an automatic transfer of valid received data by upstream DMA. The ‘Receive Data’ interrupt is used to trigger this transfer.

18.1.2.5 Other Features

The RHSB module supports a test mode for key-on test. In this test mode a simple check of the data path is possible by loop back of downstream transmission to the upstream reception.

18.1.2.6 Caution

This product supports only the driving ability with the setting of high as the output buffer characteristics of pins (driving ability) for chip select signal of the RHSB. For setting of output buffer characteristics, see **Table 2.25, PUCCn Register Contents**.

18.1.3 Block Diagram

Figure 18.1 shows a top level block diagram of the RHSB module.

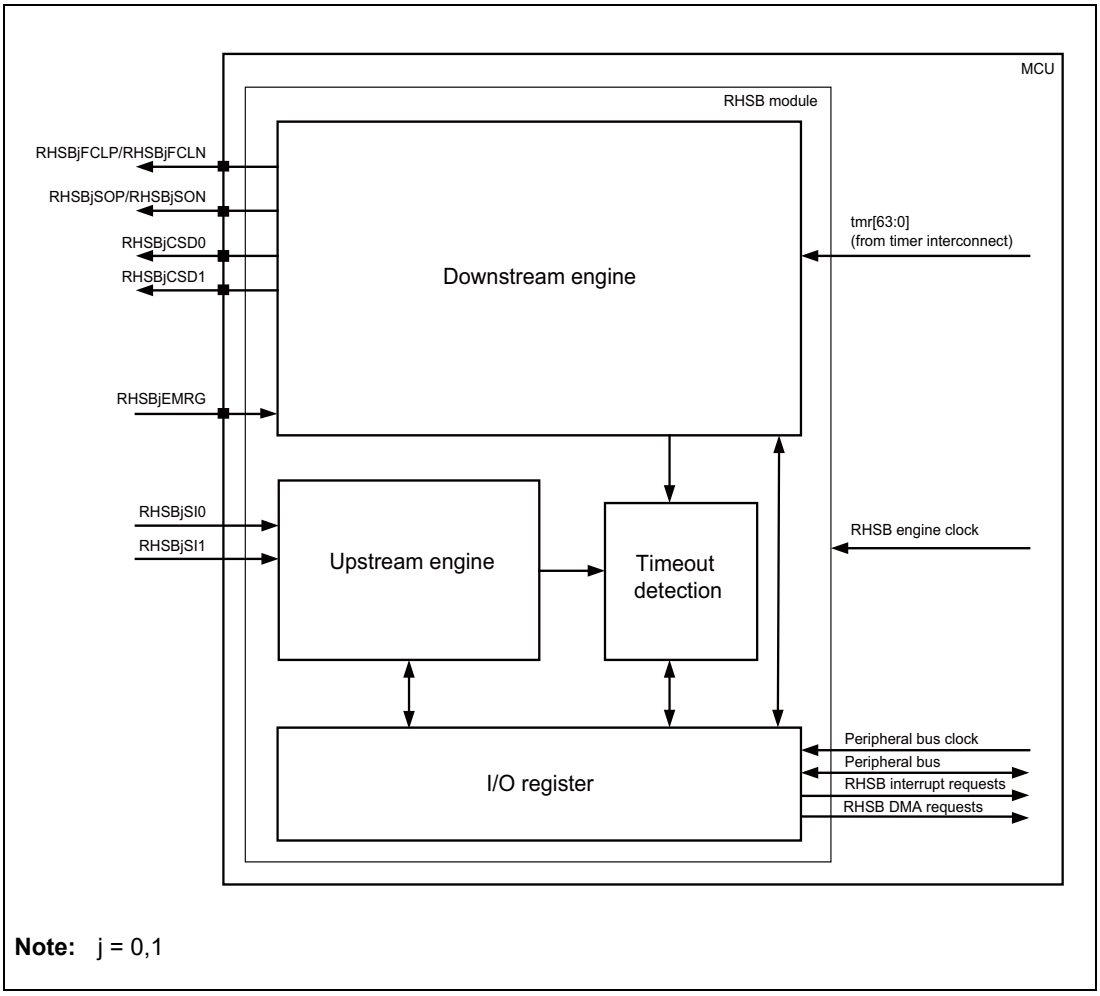


Figure 18.1 Top Level Block Diagram

The tmr[63:0] bits of the timer correspond to the ATU-IV and APA selected by the XBAR. For details of the XBAR, see **Section 18.7, Cross Bar (XBAR)**.

18.2 Register Descriptions

This section describes in detail the I/O registers of the RHSB module.

In case of a concurrent update of the same register bit, RHSB module write (hardware write) has higher priority than CPU write (software write).

18.2.1 I/O Registers Overview

The RHSB address range starts from <RHSBj_base> + 00_H and ends at <RHSBj_base> + $7F_H$.

The base address <RHSBj_base> is shown in **Table 18.4**.

The register map of a RHSB module is shown in **Table 18.5**.

Table 18.4 Base Address <RHSBj_base>

RHSBj	RHSBj_base
RHSB0	FFEE 0000_H
RHSB1	FFEE 1000_H

The RHSB module remains in reset while RHSBjGC.RHSBjOPS is RESET.

The values shown in **Table 18.5** and in the I/O registers description under 'Values after Reset' are related to this reset state.

For further information about the RHSB operation modes please refer to **Section 18.3, Operation States**.

The module specific base address of a RHSB module needs to be added to each of the specified offset addresses.

Table 18.5 I/O Registers Address Map (1/2)

Register Name	Symbol *1	Value after Reset	Address	Access Size
Global configuration register	RHSBjGC	0000 0000_H	<RHSBj_base> + 00_H	32, 16, 8
Module status register	RHSBjMSR	0001 0000_H	<RHSBj_base> + 08_H	32, 16, 8
Downstream configuration register	RHSBjDCR	0011 $FF00_H$	<RHSBj_base> + 10_H	32, 16
Data element configuration register	RHSBjDEC	0000 0000_H	<RHSBj_base> + 14_H	32, 16, 8
Slave device configuration register i (i = 0)	RHSBjSDCi	0000 0000_H	<RHSBj_base> + 18_H + 4*i	32, 16
Data element bit assignment register m (m = 0 to 3)	RHSBjDEBAm	0000 0000_H	<RHSBj_base> + 20_H + 4*m	32, 16
Emergency bit enable register i (i = 0, 1)	RHSBjEBEi	0000 0000_H	<RHSBj_base> + 30_H + 4*i	32, 16
Downstream transmission control register	RHSBjDTC	0000 0000_H	<RHSBj_base> + 38_H	32, 16, 8
Downstream command data register	RHSBjDCD	0000 0000_H	<RHSBj_base> + $3C_H$	32, 16
Downstream data register i (i = 0, 1)	RHSBjDDRi	0000 0000_H	<RHSBj_base> + 40_H + 4*i	32, 16
Downstream emergency data register i (i = 0, 1)	RHSBjDEDi	0000 0000_H	<RHSBj_base> + 48_H + 4*i	32, 16
Upstream configuration register	RHSBjUCR	0000 1800_H	<RHSBj_base> + 50_H	32, 16, 8
Upstream channel configuration register	RHSBjUCC	0F0F $0F0F_H$	<RHSBj_base> + 54_H	32, 16, 8
Upstream channel selection register	RHSBjUCS	0000 0000_H	<RHSBj_base> + 58_H	32, 16, 8
Upstream data read register	RHSBjUDR	0000 0000_H	<RHSBj_base> + $5C_H$	32, 16, 8
Upstream data register i (i = 0, 1)	RHSBjUDi	0000 0000_H	<RHSBj_base> + 60_H + 4*i	32, 16, 8
Upstream status summary register	RHSBjUSS	0000 0000_H	<RHSBj_base> + 70_H	32, 16, 8
Interrupt control register	RHSBjIC	0000 0000_H	<RHSBj_base> + 74_H	32, 16, 8
Interrupt status register	RHSBjIS	0000 0000_H	<RHSBj_base> + 78_H	32, 16, 8

Table 18.5 I/O Registers Address Map (2/2)

Register Name	Symbol *1	Value after Reset	Address	Access Size
Downstream configuration register for period 1	RHSBjDCR1	0000 0000_H	<RHSBj_base> + $7C_H$	32, 16, 8

Note 1. j = 0, 1

The access size mentioned in **Table 18.5** is obligatory for write accesses. In addition all registers are readable by 8, 16 and 32 bit access.

The user should not access the registers marked as reserved in **Table 18.5**.

The user should not write bits to a value specified as 'invalid' in the I/O registers description.

18.2.2 Legend

This section explains the module state dependent abbreviations used for the I/O registers description in **Section 18.2.3, Description of the Registers in the Common Controller** to **Section 18.2.6, Interrupt Registers**.

Conditions:

- R/W: Bit is readable and writable
- R: Read-only bit; the user cannot write to this bit

Indexes used for register and register bit names

i: Universal index

m: Index related to DFTE m

n: Index related to slave n

Reserved bits:

- Always read as 0
- Writing to these bits has no effect (hardware protection)

18.2.3 Description of the Registers in the Common Controller

18.2.3.1 RHSBjGC — Global Configuration Register

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: <RHSBj_base> + 00_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	RHSBj DCDE	RHSBj DDE	RHSBj UDE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RHSBjOPS[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Table 18.6 RHSBjGC Register Contents

Bit Position	Bit Name	Function
31 to 19	—	Reserved These bits are always read as 0. The write value should be always 0.
18	RHSBjDCDE	Downstream Command DMA Request Enable 0_B: Downstream command DMA request is disabled. 1_B: Downstream command DMA request is enabled.
17	RHSBjDDE	Downstream Data DMA Request Enable 0_B: Downstream data DMA request is disabled. 1_B: Downstream data DMA request is enabled.
16	RHSBjUDE	Upstream DMA Request Enable 0_B: Upstream DMA request is disabled. 1_B: Upstream DMA request is enabled.
15 to 2	—	Reserved These bits are always read as 0. The write value should be always 0.
1, 0	RHSBjOPS[1:0]	Operation Status 00_B: RESET state 01_B: CONFIG state 10_B: ACTIVE state 11_B: TEST state

(1) RHSBjGC.RHSBjDCDE — Downstream Command DMA Request Enable

The user should not set this bit to 1_B when RHSBjIS.RHSBjCTF is used as interrupt source (RHSBjIC.RHSBjCTIE is 1_B).

This bit defines if the downstream command data register (RHSBjDCD) and downstream transmission control register (RHSBjDTC) are updated by downstream DMA transfer and command data are transferred to slaves when RHSBjIS.RHSBjCTF is 1_B.

For details about downstream command DMA refer to **Section 18.6.2, DMA Usage for Downstream Command Transmission**.

(2) RHSBjGC.RHSBjDDE — Downstream Data DMA Enable

The user should not set this bit to 1_B when RHSBjIS.RHSBjDTSF is used as interrupt source (RHSBjIC.RHSBj DTSIE is 1_B).

This bit defines if the downstream data registers (RHSBjDDRi) are updated by downstream DMA transfer when RHSBjIS.RHSBjDTSF is 1_B.

For details about downstream data DMA refer to **Section 18.6.1, DMA Usage for Downstream Data Transmission**.

(3) RHSBjGC.RHSBjUDE — Upstream DMA Enable

The user should not set this bit to 1_B when RHSBjIS.RHSBjDRF is used as interrupt source (RHSBjIC.RHSBjDRIE is 1_B).

This bit defines if the upstream data register (RHSBjUDR) is transferred by upstream DMA when RHSBjIS.RHSBjDRF is 1_B.

For details about upstream DMA refer to **Section 18.6.3, DMA Usage for Upstream Data Reception**.

(4) RHSBjGC.RHSBjOPS — Operation Status

The user cannot write 11_B (TEST) to these bits while RHSBjGC.RHSBjOPS is 00_B or 10_B.

The user cannot write 10_B (ACTIVE) to these bits while RHSBjGC.RHSBjOPS is 00_B or 11_B.

These bits define the current operation state of the RHSB module.

Refer to **Section 18.3, Operation States**.

18.2.3.2 RHSBjMSR — Module Status Register

Access: This register can be read in 8-, 16-, or 32-bit units.

Address: <RHSBj_base> + 08_H

Value after reset: 0001 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	RHSBjDFA[2:0]			—	—	—	—	—	—	—	RHSBj TPS
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RHSBj UFA
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 18.7 RHSBjMSR Register Contents

Bit Position	Bit Name	Function
31 to 27	—	Reserved These bits are always read as 0. The write value should be always 0
26 to 24	RHSBjDFA[2:0]	Downstream Frame Activity 000_B: Inter-frame passive or disabled 001_B: Passive phase of data frame 010_B: Active phase of command frame 011_B: Passive phase of command frame 100_B: Active phase of data frame (DFTE0) 101_B: Active phase of data frame (DFTE1) 110_B: Active phase of data frame (DFTE2) 111_B: Active phase of data frame (DFTE3)
23 to 17	—	Reserved These bits are always read as 0. The write value should be always 0
16	RHSBjTPS	Transmission Period Status 0_B: Period 0 1_B: Period 1
15 to 1	—	Reserved These bits are always read as 0. The write value should be always 0
0	RHSBjUFA	Upstream Frame Activity 0_B: Upstream is idle 1_B: Upstream frame decoding is ongoing

(1) Downstream Frame Activity (RHSBjMSR.RHSBjDFA)

These bits indicate details of the downstream frame transmission. These bits are only valid when the downstream communication rate is equal to or falls below 5 megabits per second.

DTFE being transferred is flagged for the active phase of a data frame. A flag is placed in the slave select bit (assert phase or deassert phase) as a part of related DFTE.

[Changing conditions]

These bits are updated when the downstream communication phase changes.

These bits are updated when the number of DFTE used for transmission changes.

These bits are set to 000_B when entering CONFIG state.

(2) Transmission Period Status (the RHSBjTPS bit in RHSBjMSR)

This bit is only valid in multi-period repetition mode.

This bit counts and specifies the transmission period.

This bit starts to operate from 0_B when a transmission starts.

[Changing condition]

When multi-period repetition mode is selected, this bit is updated at the timing of time tick.

[Setting conditions]

When a user writes 1_B to the RHSBjDTE bit in RHSBjDTC, reading the value of this bit returns 1_B.

When the emergency function enable bit (the RHSBjEE bit in RHSBjDCR) is set to 11_B, reading the value of this bit returns 1_B.

When entering the CONFIG state, reading the value of this bit returns 1_B.

(3) Upstream Frame Activity (RHSBjMSR.RHSBjUFA)

This bit represents that the upstream decoder has detected a start bit and a frame is currently decoded.

[Clearing condition]

This bit is set to 0_B when the upstream decoder leaves RX-Active state.

[Setting condition]

This bit is set to 1_B when the upstream decoder enters RX-Active state.

18.2.4 Downstream (Tx) Registers

18.2.4.1 RHSBjDCR — Downstream Configuration Register

Access: This register can be read/written in 16- or 32-bit units.

Address: <RHSBj_base> + 10_H

Value after reset: 0011 $FF00_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RHSBjSLS[1:0]		RHSBjDFP[5:0]						RHSBjDBR[3:0]				RHSBjCIM[1:0]		RHSBj CTD	RHSBj REP[8]
Value after reset	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RHSBjREP[7:0]								RHSBj CLP	RHSBj CAC	RHSBjEE[1:0]		RHSBj EIM	RHSBj EIP	RHSBjDMS [1:0]	
Value after reset	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 18.8 RHSBjDCR Register Contents (1/2)

Bit Position	Bit Name	Function
31, 30	RHSBjSLS[1:0]	Sequence Length Setting 00_B: 1 DFTE (DFTE0) 01_B: 2 DFTEs (DFTE0, DFTE1) 10_B: 3 DFTEs (DFTE0 to DFTE2) 11_B: 4 DFTEs (DFTE0 to DFTE3)
29 to 24	RHSBjDFP[5:0]	Data Frame Passive Length Length of the data frame passive phase
23 to 20	RHSBjDBR[3:0]	Downstream Bit Rate 0_D: Invalid 1_D: $f_{DW} = f_{PE}/2$ 2_D: Invalid 3_D: $f_{DW} = f_{PE}/4$ 4_D: $f_{DW} = f_{PE}/5$ 5_D: Invalid 6_D: Invalid 7_D: $f_{DW} = f_{PE}/8$ 8_D: $f_{DW} = f_{PE}/16$ 9_D: $f_{DW} = f_{PE}/32$ 10_D: $f_{DW} = f_{PE}/64$ 11_D: $f_{DW} = f_{PE}/128$ Others: $f_{DW} = f_{PE}/256$
19, 18	RHSBjCIM[1:0]	Command Frame Insertion Method 00_B: Time-slot method 01_B: Immediate method 10_B: Best-effort method 11_B: Invalid
17	RHSBjCTD	Command Frame Transmission Delay 0_B: No restriction 1_B: Transmission of consecutive command frames prevented
16 to 8	RHSBjREP[8:0]	Repetition Period Length Length of the data frame repetition period
7	RHSBjCLP	Clock Line Phase 0_B: Change on rising edge 1_B: Change on falling edge

Table 18.8 RHSBjDCR Register Contents (2/2)

Bit Position	Bit Name	Function
6	RHSBjCAC	Clock Active Control 0_B: Clock is inactive during the passive phases 1_B: Clock is always active
5, 4	RHSBjEE[1:0]	Emergency Enable 00_B: Emergency condition detection disabled 01_B: Emergency condition detection enabled and automatic stop disabled 10_B: Invalid 11_B: Emergency condition detection enabled and automatic stop enabled
3	RHSBjEIM	Emergency Input Mode 0_B: Edge sensitive 1_B: Level sensitive
2	RHSBjEIP	Emergency Input Polarity 0_B: Active level is high / Rising edge 1_B: Active level is low / Falling edge
1, 0	RHSBjDMS[1:0]	Downstream Mode Select 00_B: Single-period repetition mode 01_B: Triggered mode 10_B: Multi-period repetition mode 11_B: Invalid

(1) Sequence Length Setting (RHSBjDCR.RHSBjSLS)

The user can write to these bits only if RHSBjGC.RHSBjOPS is 01_B (CONFIG).

These bits define the number of DFTEs used to assemble one downstream data frame.

(2) Data Frame Passive Length (RHSBjDCR.RHSBjDFP)

The user can write to these bits only if RHSBjGC.RHSBjOPS is 01_B (CONFIG).

These bits define the length of the frame passive phase (1 to 64_D bits) for data frames.

That is, when these bits are 0_D, the frame passive phase has the length of 1_D bit.

Note that a frame passive phase of 1_D bit is not valid according MSC specification.

(3) Downstream Bit Rate (RHSBjDCR.RHSBjDBR)

The user can write to these bits only if RHSBjGC.RHSBjOPS is 01_B (CONFIG).

These bits define how the downstream bit rate is derived from the f_{PE} clock.

(4) Command Frame Insertion Method (RHSBjDCR.RHSBjCIM)

The user can write to these bits only if RHSBjGC.RHSBjOPS is 01_B (CONFIG).

This bit defines how a command frame is inserted into an ongoing downstream schedule.

The following methods are available.

Time-slot method:

Command frame transmission is delayed to be handled within the defined repetition period. The repetition period is not affected by command frame transmission.

Immediate method:

Command frame transmission is handled as soon as possible. The prepetition period can be re-adjusted in case of command frame transmission.

Best-effort method:

Command frame transmission is handled as soon as possible with minimum impact to the data frame transmission. The repetition period is not affected by command frame transmission.

(5) Command Frame Transmission Delay (RHSBjDCR.RHSBjCTD)

The user can write to this bit only if RHSBjGC.RHSBjOPS is 01_B (CONFIG).

This bit defines if the command frame transmission is restricted to have at least one data frame between two command frames.

(6) Repetition Period Length (RHSBjDCR.RHSBjREP)

The user can write to these bits only if RHSBjGC.RHSBjOPS is 01_B (CONFIG).

These bits define the length of repetition period (1 to 512_D bits) used for downstream communication in single-period repetition mode and multi-period repetition mode.

That is, when these bits are 0_D the repetition period is 1_D bit.

Please refer to **Section 18.4.1.3, Downstream Modes (2) Repetition mode (Single-period repetition, Multi-period repetition)** for details.

(7) Clock Line Phase (RHSBjDCR.RHSBjCLP)

The user can write to this bit only if RHSBjGC.RHSBjOPS is 01_B (CONFIG).

This bit defines if the serial data line (RHSBjSOP, RHSBjSON (j = 0, 1)) and the chip select lines (RHSB0CSD0-1, RHSB1CSD0-1) are changing with the rising or falling edge of the serial clock line (RHSBjFCLP, RHSBjFCLN (j = 0, 1)).

(8) Clock Active Control (RHSBjDCR.RHSBjCAC)

This bit defines the activity of the downstream serial clock line (RHSBjFCLP, RHSBjFCLN (j = 0, 1)) during the passive phases. During active phases the clock is always active.

(9) Emergency Input Enable (RHSBjDCR.RHSBjEE)

The user can write to these bits only if RHSBjGC.RHSBjOPS is 01_B (CONFIG).

These bits enable the emergency function of the RHSB module.

In addition these bits define if the data transmission is stopped after an emergency frame has been transmitted.

(10) Emergency Input Mode (RHSBjDCR.RHSBjEIM)

The user can write to this bit only if RHSBjGC.RHSBjOPS is 01_B (CONFIG).

This bit defines how the emergency condition is derived from the external emergency signal (RHSBjEMRG (j = 0, 1)).

(11) Emergency Input Polarity (RHSBjDCR.RHSBjEIP)

The user can write to this bit only if RHSBjGC.RHSBjOPS is 01_B (CONFIG).

This bit defines the active level of the external emergency signal (RHSBjEMRG(j = 0, 1)).

(12) Downstream Mode Select (RHSBjDCR.RHSBjDMS)

The user can write to these bits only if RHSBjGC.RHSBjOPS is 01_B (CONFIG).

These bits define the timing mode of the downstream channel.

18.2.4.2 RHSBjDCR1 — Downstream Configuration Register for Period 1

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: <RHSBj_base> + $7C_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RHSBjSLS1[1:0]		—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 18.9 RHSBjDCR1 Register Contents

Bit Position	Bit Name	Function
31, 30	RHSBjSLS1[1:0]	Sequence Length Setting for Period 1 00_B: 1 DFTE (DFTE0) 01_B: 2 DFTEs (DFTE0, DFTE1) 10_B: 3 DFTEs (DFTE0 to DFTE2) 11_B: 4 DFTEs (DFTE0 to DFTE3)
29 to 0	—	Reserved These bits are always read as 0. The write value should be always 0

(1) Sequence Length Setting for Period 1 (RHSBjDCR1.RHSBjSLS1)

The user can write to these bits only if RHSBjGC.RHSBjOPS is 01_B (CONFIG).

These bits define the number of DFTEs used to assemble one downstream data frame.

These bits are available only in multi-period repetition mode. RHSBjDCR1.RHSBjSLS1 value should be equal to or smaller than RHSBjDCR.RHSBjSLS value.

18.2.4.3 RHSBjDEC — Data Element Configuration Register

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: <RHSBj_base> + 14_H

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	RHSBjSSD0[1:0]		RHSBjNDB0[3:0]				—	—	RHSBjSSD1[1:0]		RHSBjNDB1[3:0]			
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	RHSBjSSD2 [1:0]		RHSBjNDB2[3:0]				—	—	RHSBjSSD3 [1:0]		RHSBjNDB3[3:0]			
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Table 18.10 RHSBjDEC Register Contents

Bit Position	Bit Name	Function
31, 30	—	Reserved These bits are always read as 0. The write value should be always 0
29, 28	RHSBjSSD0 [1:0]	Slave Selected for DFTE0 Slave selected when transmitting DFTE0
27 to 24	RHSBjNDB0 [3:0]	Number of Data Bits 0 Number of data bits used from DFTE0
23, 22	—	Reserved These bits are always read as 0. The write value should be always 0
21, 20	RHSBjSSD1 [1:0]	Slave Selected for DFTE1 Slave selected when transmitting DFTE1
19 to 16	RHSBjNDB1 [3:0]	Number of Data Bits 1 Number of data bits used from DFTE1
15, 14	—	Reserved These bits are always read as 0. The write value should be always 0
13, 12	RHSBjSSD2 [1:0]	Slave Selected for DFTE2 Slave selected when transmitting DFTE2
11 to 8	RHSBjNDB2 [3:0]	Number of Data Bits 2 Number of data bits used from DFTE2
7, 6	—	Reserved These bits are always read as 0. The write value should be always 0
5, 4	RHSBjSSD3 [1:0]	Slave Selected for DFTE3 Slave selected when transmitting DFTE3
3 to 0	RHSBjNDB3 [3:0]	Number of Data Bits 3 Number of data bits used from DFTE3

(1) Slave Selected for DFTEm (RHSBjDEC.RHSBjSSDm) (m = 0 to 3)

The user can write to these bits only if RHSBjGC.RHSBjOPS is 01_B (CONFIG).

These bits define which slave is addressed by downstream frame transmission element m (DFTEm). Setting 10_B and 11_B is prohibited.

(2) Number of Data Bits m (RHSBjDEC.RHSBjNDBm) (m = 0 to 3)

The user can write to these bits only if RHSBjGC.RHSBjOPS is 01_B (CONFIG).

These bits define the number of bits (1 to 16) used from DFTEm for data frame transmission.

That is, when these bits are 0_D, 1 bit from DFTEm is transmitted.

18.2.4.4 RHSBjSDCi — Slave Device Configuration Register i (i = 0)

Access: This register can be read/written in 16- or 32-bit units.

Address: <RHSBj_base> + 18_H + 4_H × i

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	RHSBjAPLn[2:0]			—	—	RHSBj CSLPn	RHSBj SOLPn	RHSBj CPSn	RHSBjDPLn[2:0]		
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	RHSBjAPLn[2:0]			—	—	RHSBj CSLPn	RHSBj SOLPn	RHSBj CPSn	RHSBjDPLn[2:0]		
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Note 1. n = 2 * i (for bits 31 to 16)

Note 2. n = 2 * i + 1 (for bits 15 to 0)

Table 18.11 RHSBjSDCi Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 27	—	Reserved These bits are always read as 0. The write value should be always 0.
26 to 24	RHSBjAPLn[2:0]	Assertion Phase Length Assertion phase length for slave 0
23, 22	—	Reserved These bits are always read as 0. The write value should be always 0.
21	RHSBjCSLPn	Chip Select Line Polarity Chip select line polarity for slave 0 0_B: Active high 1_B: Active low
20	RHSBjSOLPn	Serial Out Line Polarity Serial out line polarity for slave 0 0_B: Inversion disabled 1_B: Inversion enabled
19	RHSBjCPSn	Content Phase Selection Bit Enable Content phase selection bit enable for slave 0 0_B: No selection bit is present 1_B: Selection bit is present
18 to 16	RHSBjDPLn[2:0]	Deassertion Phase Length Deassertion phase length for slave 0
15 to 11	—	Reserved These bits are always read as 0. The write value should be always 0.
10 to 8	RHSBjAPLn[2:0]	Assertion Phase Length Assertion phase length for slave 1
7, 6	—	Reserved These bits are always read as 0. The write value should be always 0.
5	RHSBjCSLPn	Chip Select Line Polarity Chip select line polarity for slave 1 0_B: Active high 1_B: Active low
4	RHSBjSOLPn	Serial Out Line Polarity Serial out line polarity for slave 1 0_B: Inversion disabled 1_B: Inversion enabled

Table 18.11 RHSBjSDCi Register Contents (2/2)

Bit Position	Bit Name	Function
3	RHSBjCPSn	Content Phase Selection Bit Enable Content phase selection bit enable for slave 1 0_B: No selection bit is present 1_B: Selection bit is present
2 to 0	RHSBjDPLn[2:0]	Deassertion Phase Length Deassertion phase length for slave 1

(1) Assertion Phase Length (RHSBjSDCi.RHSBjAPLn) (n = 0, 1)

The user can write to these bits only if RHSBjGC.RHSBjOPS is 01_B (CONFIG).

These bits define the assertion phase (0 to 7_D bits) used for frames transmitted to slave n. If these bits are 0_D no assertion phase is present.

(2) Chip Select Line Polarity (RHSBjSDCi.RHSBjCSLPn) (n = 0, 1)

The user can write to this bit only if RHSBjGC.RHSBjOPS is 01_B (CONFIG).

This bit defines the polarity of the chip select line (RHSBjCSDn (n = 0, 1)) for slave n.

(3) Serial Out Line Polarity (RHSBjSDCi.RHSBjSOLPn) (n = 0, 1)

The user can write to this bit only if RHSBjGC.RHSBjOPS is 01_B (CONFIG).

This bit defines the inversion of the serial data line (RHSBjSOP, RHSBjSON (j = 0, 1)) when slave n is selected.

(4) Content Phase Selection Bit enable (RHSBjSDCi.RHSBjCPSn) (n = 0, 1)

The user can write to this bit only if RHSBjGC.RHSBjOPS is 01_B (CONFIG).

This bit defines if there is a selection bit in data frames transmitted to slave n.

Note that in command frames the selection bit is always present.

(5) Deassertion Phase Length (RHSBjSDCi.RHSBjDPLn) (n = 0, 1)

The user can write to these bits only if RHSBjGC.RHSBjOPS is 01_B (CONFIG).

These bits define the deassertion phase (0 to 7_D bits) used for frames transmitted to slave n. If these bits are 0_D no deassertion phase is present.

18.2.4.5 RHSBjDEBAm — Data Element Bit Assignment Register m (m = 0 to 3)

Access: This register can be read/written in 16- or 32-bit units.

Address: <RHSBj_base> + 20_H + 4_H × m

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RHSBjDSS15 [1:0]		RHSBjDSS14 [1:0]		RHSBjDSS13 [1:0]		RHSBjDSS12 [1:0]		RHSBjDSS11 [1:0]		RHSBjDSS10 [1:0]		RHSBjDSS9 [1:0]		RHSBjDSS8 [1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RHSBjDSS7 [1:0]		RHSBjDSS6 [1:0]		RHSBjDSS5 [1:0]		RHSBjDSS4 [1:0]		RHSBjDSS3 [1:0]		RHSBjDSS2 [1:0]		RHSBjDSS1 [1:0]		RHSBjDSS0 [1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 18.12 RHSBjDEBAm Register Contents

Bit Position	Bit Name	Function
31, 30	RHSBjDSS15 [1:0]	Data Source Select 15 Data source select for bit 15 in DFTEm
29, 28	RHSBjDSS14 [1:0]	Data Source Select 14 Data source select for bit 14 in DFTEm
27, 26	RHSBjDSS13 [1:0]	Data Source Select 13 Data source select for bit 13 in DFTEm
25, 24	RHSBjDSS12 [1:0]	Data Source Select 12 Data source select for bit 12 in DFTEm
23, 22	RHSBjDSS11 [1:0]	Data Source Select 11 Data source select for bit 11 in DFTEm
21, 20	RHSBjDSS10 [1:0]	Data Source Select 10 Data source select for bit 10 in DFTEm
19, 18	RHSBjDSS9 [1:0]	Data Source Select 9 Data source select for bit 9 in DFTEm
17, 16	RHSBjDSS8 [1:0]	Data Source Select 8 Data source select for bit 8 in DFTEm
15, 14	RHSBjDSS7 [1:0]	Data Source Select 7 Data source select for bit 7 in DFTEm
13, 12	RHSBjDSS6 [1:0]	Data Source Select 6 Data source select for bit 6 in DFTEm
11, 10	RHSBjDSS5 [1:0]	Data Source Select 5 Data source select for bit 5 in DFTEm
9, 8	RHSBjDSS4 [1:0]	Data Source Select 4 Data source select for bit 4 in DFTEm
7, 6	RHSBjDSS3 [1:0]	Data Source Select 3 Data source select for bit 3 in DFTEm
5, 4	RHSBjDSS2 [1:0]	Data Source Select 2 Data source select for bit 2 in DFTEm
3, 2	RHSBjDSS1 [1:0]	Data Source Select 1 Data source select for bit 1 in DFTEm
1, 0	RHSBjDSS0 [1:0]	Data Source Select 0 Data source select for bit 0 in DFTEm

(1) Data Source Select i (RHSBjDEBAm.RHSBjDSSi) (i = 0 to 15)

The user can write to these bits only if RHSBjGC.RHSBjOPS is 01_B (CONFIG).

These bits define the data source of bit i in RHSBjDFTEm.

00_B: Bit from timer input is selected

01_B: Inverted bit from timer input is selected

10_B: Bit from downstream data register RHSBjDDR0 or RHSBjDDR1 is selected

11_B: Invalid

Please refer to **Section 18.4.1.6, Data Frame Assembling** for details about the bit selection and the bit mapping.

18.2.4.6 RHSBjEBEi — Emergency Bit Enable Register i (i = 0, 1)

Access: This register can be read/written in 16- or 32-bit units.

Address: <RHSBj_base> + 30_H + 4_H × i

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RHSBjEBE[15 + 32*i:8 + 32*i]								RHSBjEBE[7 + 32*i:32*i]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RHSBjEBE[31 + 32*i:24 + 32*i]								RHSBjEBE[23 + 32*i:16 + 32*i]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 18.13 RHSBjEBEi Register Contents

Bit Position	Bit Name	Function
31 to 24	RHSBjEBE[15 + 32*i:8 + 32*i]	Emergency Bit Enable Emergency Enable of DFTE i*2 (DFTE0, DFTE2) 0_B: Emergency function for this bit is disabled 1_B: Emergency function for this bit is enabled
23 to 16	RHSBjEBE[7 + 32*i:32*i]	
15 to 8	RHSBjEBE[31 + 32*i:24 + 32*i]	Emergency Bit Enable Emergency Enable of DFTE 1+i*2 (DFTE1, DFTE3) 0_B: Emergency function for this bit is disabled 1_B: Emergency function for this bit is enabled
7 to 0	RHSBjEBE[23 + 32*i:16 + 32*i]	

(1) Emergency Bit Enable (RHSBjEBEi.RHSBjEBE)

The user can write to these bits only if RHSBjGC.RHSBjOPS is 01_B (CONFIG).

These bits define if the emergency function is enabled for a data frame bit.

Each bit enables the emergency function for the corresponding bit in a RHSBjDFTE.RHSBjDFTE0 is controlled by RHSBjEBE0[31:16], RHSBjDFTE1 by RHSBjEBE0[15:0], RHSBjDFTE2 by RHSBjEBE1[31:16] and RHSBjDFTE3 by RHSBjEBE1[15:0].

Please refer to **Section 18.4.1.6, Data Frame Assembling** for details.

18.2.4.7 RHSBjDTC — Downstream Transmission Control Register

Do not rewrite this register using bit-manipulation instruction.

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: <RHSBj_base> + 38_H

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	RHSBjNCB[4:0]					—	—	RHSBjSSCF [1:0]		—	—	RHSBjCTR [1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	RHSBj TSR	RHSBj DTE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Table 18.14 RHSBjDTC Register Contents

Bit Position	Bit Name	Function
31 to 29	—	Reserved These bits are always read as 0. The write value should be always 0.
28 to 24	RHSBjNCB[4:0]	Number of Command Bits Number of bits used for command frame transmission
23, 22	—	Reserved These bits are always read as 0. The write value should be always 0.
21, 20	RHSBjSSCF [1:0]	Slave Selection for Command Frame 00_B: Command is send to slave 0 01_B: Command is send to slave 1 10_B: Setting prohibited 11_B: Setting prohibited.
19, 18	—	Reserved These bits are always read as 0. The write value should be always 0.
17, 16	RHSBjCTR [1:0]	Command Transmission Request 00_B: No request 01_B: Requests a command which does not requires a response from the receive slave. 10_B: Invalid 11_B: Requests a command which requires a response from the receive slave.
15 to 2	—	Reserved These bits are always read as 0. The write value should be always 0.
1	RHSBjTSR	Transmission Stop Request 0_B: No action 1_B: Request pending
0	RHSBjDTE	Data Transmission Enable 0_B: Disabled 1_B: Enabled

(1) Number of Command Bits (RHSBjDTC.RHSBjNCB)

The user can write to these bits only if RHSBjDTC.RHSBjCTR is 00_B.

The user should not write to these bits if downstream command DMA is enabled (RHSBjGC.RHSBjDCDE is 1_B).

These bits define the number of bits (1 to 32) used from RHSBjDCD register for command frame transmission.

That is, when these bits are 0_D, 1 bit from the RHSBjDCD register is transmitted.

(2) Slave Selection for Command Frame (RHSBjDTC.RHSBjSSCF)

The user can write to these bits only if RHSBjDTC.RHSBjCTR is 00_B.

The user should not write to these bits if downstream command DMA is enabled (RHSBjGC.RHSBjDCDE is 1_B).

These bits define the slave addressed for command frame transmission.

(3) Command Transmission Request (RHSBjDTC.RHSBjCTR)

The user cannot write to these bits if RHSBjGC.RHSBjOPS is 01_B (CONFIG).

The user cannot write 11_B to these bits if RHSBjUCS.RHSBjBSY is 1_B.

The user can write to these bits only if RHSBjDTC.RHSBjCTR is 00_B.

The user should not write to these bits if downstream command DMA is enabled (RHSBjGC.RHSBjDCDE is 1_B).

These bits are used to request a command frame transmission.

[Changing conditions]

These bits are set to 00_B when a command frame has been transmitted.

These bits are set to 00_B when entering the CONFIG state.

(4) Transmission Stop Request (RHSBjDTC.RHSBjTSR)

The user can only write 1_B to this bit.

The user cannot set this bit to 1_B if RHSBjDTC.RHSBjDTE is 0_B.

This bit defines if the data transmission should be stopped when there is no ongoing data frame transmission.

[Clearing conditions]

This bit is set to 0_B when data transmission is disabled (RHSBjDTC.RHSBjDTE is 0_B).

(5) Data Transmission Enable (RHSBjDTC.RHSBjDTE)

The user cannot write to this bit if RHSBjGC.RHSBjOPS is 01_B (CONFIG).

The user can only write 1_B to this bit.

This bit defines if the data frame transmission is enabled.

[Clearing conditions]

This bit is set to 0_B when there is a transmission stop request (RHSBjDTC.RHSBjTSR is 1_B) and this request has been processed.

This bit is set to 0_B when emergency condition detection and automatic stop is enabled

(RHSBjDCR.RHSBjEE is 11_B) and an emergency frame has been transmitted.

This bit is set to 0_B when the CONFIG state is entered.

18.2.4.8 RHSBjDCD — Downstream Command Data Register

Access: This register can be read/written in 16- or 32-bit units.

Address: <RHSBj_base> + $3C_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RHSBjCB[31:24]								RHSBjCB[23:16]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RHSBjCB[15:8]								RHSBjCB[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 18.15 RHSBjDCD Register Contents

Bit Position	Bit Name	Function
31 to 0	RHSBjCB[31:0]	Command Bit These bits define the data for command frame transmission

(1) Command Bit (RHSBjDCD.RHSBjCB)

The user can write to these bits only if RHSBjDTC.RHSBjCTR is 00_B.

The user should not write to these bits if downstream command DMA is enabled (RHSBjGC.RHSBjDCDE is 1_B).

These bits define the data bits used to assemble command frames.

The transmission order is LSB first.

Please refer to **Section 18.4.1.7, Command Frame Assembling** for details.

18.2.4.9 RHSBjDDRi — Downstream Data Register i (i = 0, 1)

Access: This register can be read/written in 16- or 32-bit units.

Address: <RHSBj_base> + 40_H + 4_H × i

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RHSBjDB[15 + 32*i:8 + 32*i]								RHSBjDB[7 + 32*i:32*i]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RHSBjDB[31 + 32*i:24 + 32*i]								RHSBjDB[23 + 32*i:16 + 32*i]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 18.16 RHSBjDDRi Register Contents

Bit Position	Bit Name	Function
31 to 16	RHSBjDB[15 + 32*i:32*i]	Data Bit These bits define the data of DFTE i*2 (DFTE0, DFTE2)
15 to 0	RHSBjDB[31 + 32*i:16 + 32*i]	Data Bit These bits define the data of DFTE 1+i*2 (DFTE1, DFTE3)

(1) Data Bit (RHSBjDDRi.RHSBjDB)

The user can write to these bits only if RHSBjIS.RHSBjDTSF is 1_B or only if RHSBjDTC.RHSBjDTE is 0_B.

If the user writes to these bits in other cases, write action are ignored and write data are disappeared.

The user should not write to these bits if downstream data DMA is enabled (RHSBjGC.RHSBjDDE is 1_B).

These bits define the data bits of the RHSBjDFTEs used to assemble data frames.

The transmission order of each RHSBjDFTE is LSB first.

RHSBjDFTE0 is assembled from RHSBjDDR0[31:16], RHSBjDFTE1 from RHSBjDDR0[15:0], RHSBjDFTE2 from RHSBjDDR1[31:16] and RHSBjDFTE3 from RHSBjDDR1[15:0].

Please refer to **Section 18.4.1.6, Data Frame Assembling** for details.

Whether or not writing to which bit of RHSBjDDRi can be a trigger of downstream transmission depends on RHSBjDCR.RHSBjSLS (the number of DFTE to be used).

For detail, see **Section 18.4.1.10, Data Update and Data Frame Transmission Request**.

18.2.4.10 iRHSBjDEDi — Downstream Emergency Data Register (i = 0, 1)

Access: This register can be read/written in 16- or 32-bit units.

Address: <RHSBj_base> + 48_H + 4_H × i

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RHSBjEB[15 + 32*i:8 + 32*i]								RHSBjEB[7 + 32*i:32*i]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RHSBjEB[31 + 32*i:24 + 32*i]								RHSBjEB[23 + 32*i:16 + 32*i]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 18.17 RHSBjDEDi Register Contents

Bit Position	Bit Name	Function
31 to 16	RHSBjEB[15 + 32*i:32*i]	Emergency Data Bit These bits define the emergency data of DFTE i*2 (DFTE0, DFTE2)
15 to 0	RHSBjEB[31 + 32*i:16 + 32*i]	Emergency Data Bit These bits define the emergency data of DFTE 1+i*2 (DFTE1, DFTE3)

(1) Emergency Data Bit (RHSBjDEDi.RHSBjEB)

The user can write to these bits only if RHSBjGC.RHSBjOPS is 01_B (CONFIG).

These bits define the bits of the RHSBjDFTEs used to assemble data frames under emergency condition.

RHSBjDFTE0 is assembled from RHSBjDED0[31:16], RHSBjDFTE1 from RHSBjDED0[15:0], RHSBjDFTE2 from RHSBjDED1[31:16] and RHSBjDFTE3 from RHSBjDED1[15:0].

Please refer to **Section 18.4.1.6, Data Frame Assembling** for details.

18.2.5 Upstream (Rx) Registers

18.2.5.1 RHSBjUCR — Upstream Configuration Register

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: <RHSBj_base> + 50_H

Value after reset: 0000 1800_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	RHSBjRTO[4:0]					—	—	—	—	RHSBj UE	RHSBj TOE	RHSBj UMS	RHSBj FSM
Value after reset	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Table 18.18 RHSBjUCR Register Contents

Bit Position	Bit Name	Function
31 to 13	—	Reserved These bits are always read as 0. The write value should be always 0.
12 to 8	RHSBjRTO[4:0]	Receive Timeout Period Timeout period in U bit
7 to 4	—	Reserved These bits are always read as 0. The write value should be always 0.
3	RHSBjUE	Upstream Enable 0_B: Disabled 1_B: Enabled
2	RHSBjTOE	Timeout Enable 0_B: Disabled 1_B: Enabled
1	RHSBjUMS	Upstream Mode Select 0_B: Dedicated mode 1_B: Shared mode
0	RHSBjFSM	Frame Storing Method 0_B: Normal storing method 1_B: Addressed storing method For details, refer to **Section 18.4.2.4, Frame Storing**.

(1) Receive Timeout Period (RHSBjUCR.RHSBjRTO)

These bits define the number of nominal upstream bit times before the timeout counter expires.

That is, when these bits are 24_D (18_H), the timeout period has a length of 25 U bits.

(2) Upstream Enable (RHSBjUCR.RHSBjUE)

The user can write to this bit only if RHSBjGC.RHSBjOPS is 01_B (CONFIG).

This bit defines if reception on the upstream channels is enabled.

(3) Timeout Enable (RHSBjUCR.RHSBjTOE)

The user can write to this bit only if RHSBjGC.RHSBjOPS is 01_B (CONFIG).

This bit defines if the timeout function is enabled.

(4) Upstream Mode Select (RHSBjUCR.RHSBjUMS)

The user can write to this bit only if RHSBjGC.RHSBjOPS is 01_B (CONFIG).

This bit defines how the slaves are connected to the serial input lines of the RHSB module.

(5) Frame Storing Method (RHSBjUCR.RHSBjFSM)

The user can write to this bit only if RHSBjGC.RHSBjOPS is 01_B (CONFIG).

This bit defines where the result of upstream decoding is stored.

18.2.5.2 RHSBjUCC — Upstream Channel Configuration Register

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: <RHSBj_base> + 54_H

Value after reset: 0F0F 0F0F$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RHSBj UFT0	RHSBj SBN0	RHSBj ILP0	RHSBj PC0	RHSBjUBR0[3:0]				RHSBj UFT1	RHSBj SBN1	RHSBj ILP1	RHSBj PC1	RHSBjUBR1[3:0]			
Value after reset	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 18.19 RHSBjUCC Register Contents

Bit Position	Bit Name	Function
31	RHSBjUFT0	Upstream Frame Type Defines if upstream frame on channel 0 uses an 8-bit or 12-bit data format
30	RHSBjSBN0	Stop Bit Number Defines if upstream frame on channel 0 uses 2 or 3 stop bit format
29	RHSBjILP0	Serial Input Line Polarity Defines if the serial input line of channel 0 (RHSBjSI0) is inverted
28	RHSBjPC0	Parity Control Defines if upstream frame on channel 0 uses even or odd parity
27 to 24	RHSBjUBR0 [3:0]	Upstream Bit Rate Defines the upstream bit rate used for channel 0
23	RHSBjUFT1	Upstream Frame Type Defines if upstream frame on channel 1 uses an 8-bit or 12-bit data format
22	RHSBjSBN1	Stop Bit Number Defines if upstream frame on channel 1 uses 2 or 3 stop bit format
21	RHSBjILP1	Serial Input Line Polarity Defines if the serial input line of channel 1 (RHSBjSI1) is inverted
20	RHSBjPC1	Parity Control Defines if upstream frame on channel 1 uses even or odd parity
19 to 16	RHSBjUBR1 [3:0]	Upstream Bit Rate Defines the upstream bit rate used for channel 1
15 to 0	—	Reserved These bits are always read as their initial values. The write value should always be the initial one.

(1) RHSBjUCC.RHSBjUFTn — Upstream Frame Type

The user can write to this bit only if RHSBjGC.RHSBjOPS is 01_B (CONFIG).

This bit defines the number of data bits of an upstream frame received on channel n.

0_B: 8-bit data field format

1_B: 12-bit data field format (including 4 bit address)

(2) RHSBjUCC.RHSBjSBNn — Stop Bit Number

The user can write to this bit only if RHSBjGC.RHSBjOPS is 01_B (CONFIG).

This bit defines the number of stop bits of an upstream frame received on channel n.

0_B: 2 stop bits format

1_B: 3 stop bits format

(3) RHSBjUCC.RHSBjILPn — Serial Input Line Polarity

The user can write to this bit only if RHSBjGC.RHSBjOPS is 01_B (CONFIG).

This bit defines if two is an inversion of the serial input line of channel n.

0_B: Inversion disabled

1_B: Inversion enabled

Note when RHSBjUCR.RHSBjUMS is shared mode, RHSBjILP0 is used for all four channels (RHSBjILP1 are ignored).

(4) RHSBjUCC.RHSBjPCn — Parity Control

The user can write to this bit only if RHSBjGC.RHSBjOPS is 01_B (CONFIG).

This bit defines the parity bit interpretation of an upstream frame received on channel n.

0_B: Even parity

1_B: Odd parity

(5) RHSBjUCC.RHSBjUBRn — Upstream Bit Rate

The user can write to these bits only if RHSBjGC.RHSBjOPS is 01_B (CONFIG).

These bits define the relation from upstream to downstream bit rate for channel n.

0_D: Setting prohibited

1_D: Setting prohibited

2_D: Setting prohibited

3_D: $f_{UP} = f_{DW}/8$

4_D: $f_{UP} = f_{DW}/16$

5_D: $f_{UP} = f_{DW}/32$

6_D: $f_{UP} = f_{DW}/64$

7_D: $f_{UP} = f_{DW}/128$

8_D: $f_{UP} = f_{DW}/256$

Others: $f_{UP} = f_{DW}/512$

Not all combinations of upstream and downstream bit rate are valid. Please refer to **Section 18.4.2.5, Upstream Bit Rates** for details.

18.2.5.3 RHSBjUCS — Upstream Channel Selection Register

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: <RHSBj_base> + 58_H

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RHSBj BSY	—	—	—	—	—	RHSBjACC[1:0]		—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 18.20 RHSBjUCS Register Contents

Bit Position	Bit Name	Function
31	RHSBjBSY	Busy 0_B: Configuration ready 1_B: Processing configuration change
30 to 26	—	Reserved These bits are always read as 0. The write value should be always 0.
25, 24	RHSBjACC[1:0]	Active Channel Configuration 00_B: Configuration given for channel 0 01_B: Configuration given for channel 1 10_B: Setting prohibited 11_B: Setting prohibited
23 to 0	—	Reserved These bits are always read as 0. The write value should be always 0.

(1) RHSBjUCS.RHSBjBSY — Busy

This bit represents that the RHSB module is processing the change of active channel configuration.

[Clearing conditions]

This bit is set to 0_B when the change of active channel configuration has been processed.

This bit is set to 0_B when entering the CONFIG state.

[Setting conditions]

This bit is set to 1_B when RHSBjUCS.RHSBjACC changes and RHSBjGC.RHSBjOPS is not 01_B (CONFIG).

This bit is set to 1_B when leaving the CONFIG state.

(2) RHSBjUCS.RHSBjACC — Active Channel Configuration

The user cannot write to these bits if RHSBjUCS.RHSBjBYS is 1_B.

The user should not write to these bits after starting a command transmission with upstream data request (by setting RHSBjDTC.RHSBjCTR to 11_B) until the upstream reception has been completed. The exact period where an upstream reception is expected, is application specific.

These bits define the channel configuration specified in RHSBjUCC register to be used for upstream reception. In dedicated mode (RHSBjUCR.RHSBjUMS = 0_B) these bits also define the serial input line.

Note that an ongoing reception is aborted and the timeout detection is deactivated when these bits are changing.

[Changing condition]

These bits are updated to the value of RHSBjDTC.RHSBjSSCF when the user requests a command transmission with upstream data request by setting RHSBjDTC.RHSBjCTR to 11_B.

18.2.5.4 RHSBjUDR — Upstream Data Read Register

Access: This register can be read in 8-, 16-, or 32-bit units.

Address: <RHSBj_base> + $5C_H$

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RHSBjNDS[0:1]		—	—	—	—	RHSBjLUD[1:0]		—	—	—	RHSBj DL	RHSBj TO	RHSBj FERR	RHSBj PERR	RHSBj ND
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	RHSBjEDF[3:0]				RHSBjDF[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 18.21 RHSBjUDR Register Contents

Bit Position	Bit Name	Function
31	RHSBjNDS0	New Data Summary 0_B: No new reception of a fault free frame on upstream channel 0 1_B: New frame received without fault on upstream channel 0
30	RHSBjNDS1	New Data Summary 0_B: No new reception of a fault free frame on upstream channel 1 1_B: New frame received without fault on upstream channel 1
29 to 26	—	Reserved These bits are always read as 0. The write value should be always 0.
25, 24	RHSBjLUD[1:0]	Last Updated Data 00_B: UD0 (channel 0) 01_B: UD1 (channel 1)
23 to 21	—	Reserved These bits are always read as 0. The write value should be always 0.
20	RHSBjDL	Data Lost 0_B: No data lost situation detected 1_B: Data lost situation detected
19	RHSBjTO	Timeout Detected 0_B: No timeout situation detected 1_B: Timeout situation detected
18	RHSBjFERR	Frame Error 0_B: No stop bit error detected 1_B: Stop bit error detected
17	RHSBjPERR	Parity Error 0_B: No parity error detected 1_B: Parity error detected
16	RHSBjND	New Data 0_B: No new reception of a fault free frame 1_B: New frame received without fault
15 to 12	—	Reserved These bits are always read as 0. The write value should be always 0.
11 to 8	RHSBjEDF[3:0]	Extended Data Field 4-bit extended data field of last received frame
7 to 0	RHSBjDF[7:0]	Data Field 8-bit data field of last received frame

The user should not read the RHSBjUDR register when upstream DMA is enabled (RHSBjGC.RHSBjUDE is 1_B).

Note that reading this register sets RHSBjIS.RHSBjDRF to 0_B and sets RHSBjUDi.RHSBjND of channel i to 0_B. The channel number i is shown by RHSBjUDR.RHSBjLUD.

(1) RHSBjUDR.RHSBjNDS — New Data Summary

These bits represent the value of RHSBjUDi.RHSBjND from channel i (i = 0, 1).

[Changing condition]

This bit is updated when RHSBjUDi.RHSBjND changes.

(2) RHSBjUDR.RHSBjLUD — Last Updated Data

These bits represent the number of Upstream Data register (RHSBjUDi) where the last data update occurred.

[Changing condition]

This bit is updated when the set condition of one of the RHSBjUDi.RHSBjND bits is fulfilled.

(3) RHSBjUDR.RHSBjDL — Data Lost

This bit represents the value of RHSBjUDi.RHSBjDL from channel i. The channel number i is shown by RHSBjUDR.RHSBjLUD.

[Changing condition]

This bit is updated when RHSBjUDR.RHSBjLUD or RHSBjUDi.RHSBjDL (with i = RHSBjUDR.RHSBjLUD) change.

(4) RHSBjUDR.RHSBjTO — Timeout Detected

This bit represents the value of RHSBjUDi.RHSBjTO from channel i. The channel number i is shown by RHSBjUDR.RHSBjLUD.

[Changing condition]

This bit is updated when RHSBjUDR.RHSBjLUD or RHSBjUDi.RHSBjTO (with i = RHSBjUDR.RHSBjLUD) change.

(5) RHSBjUDR.RHSBjFERR — Frame Error

This bit represents the value of RHSBjUDi.RHSBjFERR from channel i. The channel number i is shown by RHSBjUDR.RHSBjLUD.

[Changing condition]

This bit is updated when RHSBjUDR.RHSBjLUD or RHSBjUDi.RHSBjFERR (with i = RHSBjUDR.RHSBjLUD) change.

(6) RHSBjUDR.RHSBjPERR — Parity Error

This bit represents the value of RHSBjUDi.RHSBjPERR from channel i. The channel number i is shown by RHSBjUDR.RHSBjLUD.

[Changing condition]

This bit is updated when RHSBjUDR.RHSBjLUD or RHSBjUDi.RHSBjPERR (with i = RHSBjUDR.RHSBjLUD) change.

(7) RHSBjUDR.RHSBjND — New Data

This bit represents the value of RHSBjUDi.RHSBjND from channel i. The channel number i is shown by RHSBjUDR.RHSBjLUD.

[Changing condition]

This bit is updated when RHSBjUDR.RHSBjLUD or RHSBjUDi.RHSBjND (with i = RHSBjUDR.RHSBjLUD) change.

(8) RHSBjUDR.RHSBjEDF — Extended Data Field

This bit represents the value of RHSBjUDi.RHSBjEDF from channel i. The channel number i is shown by RHSBjUDR.RHSBjLUD.

[Changing condition]

This bit is updated when RHSBjUDR.RHSBjLUD or RHSBjUDi.RHSBjEDF (with i = RHSBjUDR.RHSBjLUD) change.

(9) RHSBjUDR.RHSBjDF — Data Field

This bit represents the value of RHSBjUDi.RHSBjDF from channel i. The channel number i is shown by RHSBjUDR.RHSBjLUD.

[Changing condition]

This bit is updated when RHSBjUDR.RHSBjLUD or RHSBjUDi.RHSBjDF (with i = RHSBjUDR.RHSBjLUD) change.

18.2.5.5 RHSBjUDi — Upstream Data Register i (i = 0, 1)

Do not clear the RHSBjDL, RHSBjTO, RHSBjFERR, RHSBjPERR, and RHSBjND bits in this register using bit-manipulation instruction.

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: <RHSBj_base> + 60_H + 4_H × i

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	RHSBj DL	RHSBj TO	RHSBj FERR	RHSBj PERR	RHSBj ND
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	RHSBjEDF[3:0]				RHSBjDF[7:0]							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 18.22 RHSBjUDi Register Contents

Bit Position	Bit Name	Function
31 to 21	—	Reserved These bits are always read as 0. The write value should be always 0.
20	RHSBjDL	Data Lost 0_B: No data lost situation detected 1_B: Data lost situation detected
19	RHSBjTO	Timeout Detected 0_B: No timeout situation detected 1_B: Timeout situation detected
18	RHSBjFERR	Frame Error 0_B: No stop bit error detected 1_B: Stop bit error detected
17	RHSBjPERR	Parity Error 0_B: No parity error detected 1_B: Parity error detected
16	RHSBjND	New Data 0_B: No valid data available 1_B: Valid data available
15 to 12	—	Reserved These bits are always read as 0. The write value should be always 0.
11 to 8	RHSBjEDF[3:0]	Extended Data Field 4-bit extended data field of received frame
7 to 0	RHSBjDF[7:0]	Data Field 8-bit data field of received frame

(1) RHSBjUDi.RHSBjDL — Data Lost

The user can only write 0_B to this bit.

This bit represents that a fault free frame was received on channel i while the RHSBjND bit in this register is 1_B.

[Clearing condition]

This bit is set to 0_B when entering the CONFIG state.

[Setting condition]

This bit is set to 1_B when the set condition of ND fulfilled while the ND bit in this register is 1_B.

(2) RHSBjUDi.RHSBjTO — Timeout Detected

The user can only write 0_B to this bit.

This bit represents that there was no valid reception, after a command frame transmission with upstream data request, within the timeout period.

[Clearing condition]

This bit is set to 0_B when entering the CONFIG state.

[Setting condition]

This bit is set to 1_B when the timeout counter expires and RHSBjUCS.RHSBjACC is indexing this channel.

(3) RHSBjUDi.RHSBjFERR — Frame Error

The user can only write 0_B to this bit.

This bit represents that a frame with stop bit error was received on channel i.

[Clearing condition]

This bit is set to 0_B when entering the CONFIG state.

[Setting condition]

This bit is set to 1_B after an upstream reception with at least one faulty stop bit.

(4) RHSBjUDi.RHSBjPERR — Parity Error

The user can only write 0_B to this bit.

This bit represents that a frame with parity error was received on channel i.

[Clearing condition]

This bit is set to 0_B when entering the CONFIG state.

[Setting condition]

This bit is set to 1_B after an upstream reception with faulty parity bit.

(5) RHSBjUDi.RHSBjND — New Data

The user can only write 0_B to this bit.

This bit represents that RHSBjUDi.RHSBjEDF and RHSBjUDi.RHSBjDF containing valid data received on channel i.

[Clearing conditions]

This bit is set to 0_B when there is a read access to the RHSBjUDR and RHSBjUDR.RHSBjLUD is indexing this channel.

This bit is set to 0_B when entering the CONFIG state.

[Setting condition]

This bit is set to 1_B after an upstream reception without parity and without stop bit errors.

(6) RHSBjUDi.RHSBjEDF — Extended Data Field

These bits represent the 4-bit extended data field bits of received message.

When using an 8-bit data format, these bits are undefined.

[Changing condition]

These bits are updated when the set condition of RHSBjND is fulfilled.

(7) RHSBjUDi.RHSBjDF — Data Field

These bits represent the 8 bit data field of received message.

[Changing condition]

These bits are updated when the set condition of RHSBjND is fulfilled.

18.2.5.6 RHSBjUSS — Upstream Status Summary Register

Access: This register can be read in 8-, 16-, or 32-bit units.

Address: <RHSBj_base> + 70_H

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	RHSBj DL0	RHSBj TO0	RHSBj FERR0	RHSBj PERR0	RHSBj ND0	—	—	—	RHSBj DL1	RHSBj TO1	RHSBj FERR1	RHSBj PERR1	RHSBj ND1
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 18.23 RHSBjUSS Register Contents

Bit Position	Bit Name	Function
31 to 29	—	Reserved These bits are always read as 0. The write value should be always 0.
28	RHSBjDL0	Data Lost Data lost indication of channel 0 (RHSBjUD0.RHSBjDL).
27	RHSBjTO0	Timeout Detected Time out indication of channel 0 (RHSBjUD0.RHSBjTO).
26	RHSBjFERR0	Frame Error Frame error indication of channel 0 (RHSBjUD0.RHSBjFERR).
25	RHSBjPERR0	Parity Error Parity error indication of channel 0 (RHSBjUD0.RHSBjPERR).
24	RHSBjND0	New Data New data indication of channel 0 (RHSBjUD0.RHSBjND).
23 to 21	—	Reserved These bits are always read as 0. The write value should be always 0.
20	RHSBjDL1	Data Lost Data lost indication of channel 1 (RHSBjUD1.RHSBjDL).
19	RHSBjTO1	Timeout Detected Timeout indication of channel 1 (RHSBjUD1.RHSBjTO).
18	RHSBjFERR1	Frame Error Frame error indication of channel 1 (RHSBjUD1.RHSBjFERR).
17	RHSBjPERR1	Parity Error Parity error indication of channel 1 (RHSBjUD1.RHSBjPERR).
16	RHSBjND1	New Data New data indication of channel 1 (RHSBjUD1.RHSBjND).
15 to 13	—	Reserved These bits are always read as 0. The write value should be always 0.
12 to 8	—	Reserved These bits are always read as 0.
7 to 5	—	Reserved These bits are always read as 0. The write value should be always 0.
4 to 0	—	Reserved These bits are always read as 0.

(1) RHSBjUSS.RHSBjDLi — Data Lost

This bit represents the value of RHSBjUDi.RHSBjDL from channel i.

0_B: No data lost situation detected

1_B: Data lost situation detected

[Changing condition]

This bit is updated when RHSBjUDi.RHSBjDL changes.

(2) RHSBjUSS.RHSBjTOi — Timeout Detected

This bit represents the value of RHSBjUDi.RHSBjTO from channel i.

0_B: No timeout situation detected

1_B: Timeout situation detected

[Changing condition]

This bit is updated when RHSBjUDi.RHSBjTO changes.

(3) RHSBjUSS.RHSBjFERRi — Frame Error

This bit represents the value of RHSBjUDi.RHSBjFERR from channel i.

0_B: No stop bit error detected

1_B: Stop bit error detected

[Changing condition]

This bit is updated when RHSBjUDi.RHSBjFERR changes.

(4) RHSBjUSS.RHSBjPERRi — Parity Error

This bit represents the value of RHSBjUDi.RHSBjPERR from channel i.

0_B: No parity error detected

1_B: Parity error detected

[Changing condition]

This bit is updated when RHSBjUDi.RHSBjPERR changes.

(5) RHSBjUSS.RHSBjNDi — New Data

This bit represents the value of RHSBjUDi.RHSBjND from channel i.

0_B: No valid data available

1_B: Valid data available

[Changing condition]

This bit is updated when RHSBjUDi.RHSBjND changes.

18.2.6 Interrupt Registers

18.2.6.1 RHSBjIC — Interrupt Control Register

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: <RHSBj_base> + 74_H

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	RHSBj DLIE	RHSBj TOIE	RHSBj UEIE	RHSBj DRIE	—	—	—	RHSBj TSIE	RHSBj ETIE	RHSBj CTIE	RHSBj DTIE	RHSBj DTSIE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 18.24 RHSBjIC Register Contents

Bit Position	Bit Name	Function
31 to 28	—	Reserved These bits are always read as 0. The write value should be always 0.
27	RHSBjDLIE	Data Lost Interrupt Enable 0_B: Disabled 1_B: Enabled
26	RHSBjTOIE	Timeout Detected Interrupt Enable 0_B: Disabled 1_B: Enabled
25	RHSBjUEIE	Upstream Error Interrupt Enable 0_B: Disabled 1_B: Enabled
24	RHSBjDRIE	Data Received Interrupt Enable 0_B: Disabled 1_B: Enabled
23 to 21	—	Reserved These bits are always read as 0. The write value should be always 0.
20	RHSBjTSIE	Transmission Started Interrupt Enable 0_B: Disabled 1_B: Enabled
19	RHSBjETIE	Emergency Frame Transmission Done Interrupt Enable 0_B: Disabled 1_B: Enabled
18	RHSBjCTIE	Command Frame Transmission Done Interrupt Enable 0_B: Disabled 1_B: Enabled
17	RHSBjDTIE	Data Frame Transmission Done Interrupt Enable 0_B: Disabled 1_B: Enabled
16	RHSBjDTSIE	Data Frame Transmission Started Interrupt Enable 0_B: Disabled 1_B: Enabled
15 to 0	—	Reserved These bits are always read as 0. The write value should be always 0.

(1) RHSBjIC.RHSBjDLIE — Data Lost Interrupt Enable

While this bit is 1_B, an interrupt will be generated if RHSBjIS.RHSBjDLF is 1_B.

(2) RHSBjIC.RHSBjTOIE — Timeout Detected Interrupt Enable

While this bit is 1_B, an interrupt will be generated if RHSBjIS.RHSBjTOF is 1_B.

(3) RHSBjIC.RHSBjUEIE — Upstream Error Interrupt Enable

While this bit is 1_B, an interrupt will be generated if RHSBjIS.RHSBjUEF is 1_B.

(4) RHSBjIC.RHSBjDRIE — Data Received Interrupt Enable

The user should not set this bit to 1_B when the upstream DMA support is enabled (RHSBjGC.RHSBjUDE is 1_B).

While this bit is 1_B, an interrupt will be generated if RHSBjIS.RHSBjDRF is 1_B.

(5) RHSBjIC.RHSBjTSIE — Transmission Started Interrupt Enable

While this bit is 1_B, an interrupt will be generated if RHSBjIS.RHSBjTSF is 1_B.

(6) RHSBjIC.RHSBjETIE — Emergency Frame Transmission Done Interrupt Enable

While this bit is 1_B, an interrupt will be generated if RHSBjIS.RHSBjETF is 1_B.

(7) RHSBjIC.RHSBjCTIE — Command Frame Transmission Done Interrupt Enable

The user should not set this bit to 1_B when the downstream command DMA transfer is enabled (RHSBjGC.RHSBjDCDE is 1_B).

While this bit is 1_B, an interrupt will be generated if RHSBjIS.RHSBjCTF is 1_B.

(8) RHSBjIC.RHSBjDTIE — Data Frame Transmission Done Interrupt Enable

While this bit is 1_B, an interrupt will be generated if RHSBjIS.RHSBjDTF is 1_B.

(9) RHSBjIC.RHSBjDTSIE — Data Frame Transmission Started Interrupt Enable

The user should not set this bit to 1_B when the downstream data DMA support is enabled (RHSBjGC.RHSBjDDE is 1_B).

While this bit is 1_B, an interrupt will be generated if RHSBjIS.RHSBjDTSF is 1_B.

18.2.6.2 RHSBjIS — Interrupt Status Register

Do not clear each bit in this register using bit-manipulation instruction

Access: This register can be read/written in 8-, 16-, or 32-bit units.

Address: <RHSBj_base> + 78_H

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	RHSBj DLF	RHSBj TOF	RHSBj UE	RHSBj DRF	—	—	—	RHSBj TSF	RHSBj ETF	RHSBj CTF	RHSBj DTF	RHSBj DTSF
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	RHSBj ERF	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R

Table 18.25 RHSBjIS Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 28	—	Reserved These bits are always read as 0. The write value should be always 0.
27	RHSBjDLF	Data Lost Flag 0_B: No data lost situation detected 1_B: Data lost situation detected
26	RHSBjTOF	Timeout Detected Flag 0_B: No timeout situation detected 1_B: Timeout situation detected
25	RHSBjUEF	Upstream Error Flag 0_B: No receive error detected 1_B: Receive error detected
24	RHSBjDRF	Data Received Flag 0_B: No reception of new valid data 1_B: Reception of new valid data
23 to 21	—	Reserved These bits are always read as 0. The write value should be always 0.
20	RHSBjTSF	Transmission Started Flag 0_B: No transmission started 1_B: Transmission has been started
19	RHSBjETF	Emergency Frame Transmission Done Flag 0_B: No emergency transmission 1_B: Emergency frame has been transmitted
18	RHSBjCTF	Command Frame Transmission Done Flag 0_B: No command frame transmission 1_B: Command frame has been transmitted
17	RHSBjDTF	Data Frame Transmission Done Flag 0_B: No data frame transmission 1_B: Data frame has been transmitted
16	RHSBjDTSF	Data Frame Transmission Started Flag 0_B: No data transmission has been started 1_B: Data transmission has been started
15 to 9	—	Reserved These bits are always read as 0. The write value should be always 0.

Table 18.25 RHSBjIS Register Contents (2/2)

Bit Position	Bit Name	Function
8	RHSBjERF	Emergency Signal Rising Flag 0_B No edge detected 1_B Edge detected
7 to 0	—	Reserved These bits are always read as 0. The write value should be always 0.

(1) RHSBjIS.RHSBjDLF — Data Lost Flag

The user can only write 0_B to this bit.

This bit represents that data lost has been detected on at least one channel.

[Clearing conditions]

This bit is set to 0_B when all RHSBjUDi.RHSBjDL (i = 0, 1) flags are 0_B.

This bit is set to 0_B when entering the CONFIG state.

[Setting condition]

This bit is set to 1_B when the set condition of at least one of these flags (RHSBjUDi.RHSBjDL with i = 0, 1) is fulfilled.

(2) RHSBjIS.RHSBjTOF — Timeout Detected Flag

The user can only write 0_B to this bit.

This bit represents that there was no valid reception requested by a command frame transmission within the timeout period.

[Clearing condition]

This bit is set to 0_B when entering the CONFIG state.

[Setting condition]

This bit is set to 1_B when the timeout counter expires.

(3) RHSBjIS.RHSBjUEF — Upstream Error Flag

The user can only write 0_B to this bit.

This bit represents that a frame with parity or stop bit error was received.

[Clearing conditions]

This bit is set to 0_B when all RHSBjUDi.RHSBjFERR (i = 0, 1) and all RHSBjUDi.RHSBjPERR (i = 0, 1) flags are 0_B.

This bit is set to 0_B when entering the CONFIG state.

[Setting conditions]

This bit is set to 1_B when the set condition of at least one of these flags (RHSBjUDi.RHSBjFERR with i = 0, 1) is fulfilled.

This bit is set to 1_B when the set condition of at least one of these flags (RHSBjUDi.RHSBjPERR with i = 0, 1) is fulfilled.

(4) RHSBjIS.RHSBjDRF — Data Received Flag

The user can only write 0_B to this bit.

The user should not set this bit to 0_B when upstream DMA is enabled (RHSBjGC.RHSBjUDE is 1_B).

This bit represents that at least one RHSBjUDi register is updated with new valid data.

[Clearing conditions]

This bit is set to 0_B when all RHSBjUDi.RHSBjND (i = 0, 1) flag are 0_B.

This bit is set to 0_B when there is a read access to RHSBjUDR.

This bit is set to 0_B when entering the CONFIG state.

[Setting condition]

This bit is set to 1_B when the set condition of at least one of these flags (RHSBjUDi.RHSBjND with i = 0, 1) is fulfilled.

(5) RHSBjIS.RHSBjTSF — Transmission Started Flag

The user can only write 0_B to this bit.

This bit indicates that a command or data frame transmission has been started.

[Clearing condition]

This bit is set to 0_B when entering the CONFIG state.

[Setting condition]

This bit is set to 1_B when a new frame transmission starts.

(6) RHSBjIS.RHSBjETF — Emergency Frame Transmission Done Flag

The user can only write 0_B to this bit.

This bit indicates that a data frame assembled under emergency condition has been transmitted.

[Clearing condition]

This bit is set to 0_B when entering the CONFIG state.

[Setting condition]

This bit is set to 1_B when an emergency frame has been transmitted.

(7) RHSBjIS.RHSBjCTF — Command Frame Transmission Done Flag

The user can only write 0_B to this bit.

The user should not set this bit to 0_B when downstream command DMA transfer is enabled (RHSBjGC.RHSBjDCDE is 1_B).

This bit indicates that a command frame has been transmitted.

[Clearing conditions]

This bit is set to $0_B$when a new command frame transmission is requested.

This bit is set to $0_B$when entering the CONFIG state.

[Setting condition]

This bit is set to 1_B when a command frame has been transmitted.

(8) RHSBjIS.RHSBjDTF — Data Frame Transmission Done Flag

The user can only write 0_B to this bit.

This bit indicates that a data frame has been transmitted.

[Clearing conditions]

This bit is set to 0_B when last active portions of the downstream data registers (RHSBjDDRi) are written (please refer to **Section 18.4.1.10, Data Update and Data Frame Transmission Request**).

This bit is set to 0_B when entering the CONFIG state.

[Setting condition]

This bit is set to 1_B when a data frame has been transmitted.

(9) RHSBjIS.RHSBjDTSF — Data Frame Transmission Started Flag

The user can only write 0_B to this bit.

The user should not set this bit to 0_B when downstream data DMA is enabled (RHSBjGC.RHSBjDDE is 1_B).

This bit indicates that a data frame transmission has been started.

When this bit is 0_B, data frame transmission is disabled or the downstream data registers (RHSBjDDRi) contain pending transmit data.

When this bit is 1_B, data frame transmission has been started. The downstream data registers can be updated to define the data for the next data frame.

[Clearing conditions]

This bit is set to 0_B when last active portions of the downstream data registers (RHSBjDDRi) are written (please refer to **Section 18.4.1.10, Data Update and Data Frame Transmission Request**).

This bit is set to 0_B when data frame transmission gets disabled (RHSBjDTC.RHSBjDTE changes from 1_B to 0_B).

This bit is set to 0_B when entering the CONFIG state.

[Setting conditions]

This bit is set to 1_B when a data frame transmission starts.

This bit is set to 1_B when data frame transmission gets enabled (RHSBjDTC.RHSBjDTE changes from 0_B to 1_B) and RHSBjDCR.RHSBjDMS is 01_B (TRIGGERED).

(10) RHSBjIS.RHSBjERF — Emergency Signal Rising Flag

The user can only write 0_B to this bit.

This bit indicates that an edge to the active level of the external emergency signal (RHSBjEMRG (j = 0, 1)) has been detected.

[Clearing condition]

This bit is set to 0_B when entering the CONFIG state.

[Setting conditions]

This bit is set to 1_B when a transition from inactive to active level of the emergency signal has been detected while RHSBjGC.RHSBjOPS is ACTIVE or TEST.

This bit is set to 1_B when there is an active level of the emergency signal when leaving the CONFIG state.

18.3 Operation States

After the release of MCU's hardware reset, the RHSB module is in RESET state. The user can trigger changes of the operation state by writing the target state to RHSBjGC.RHSBjOPS.

The current operation state is indicated by RHSBjGC.RHSBjOPS.

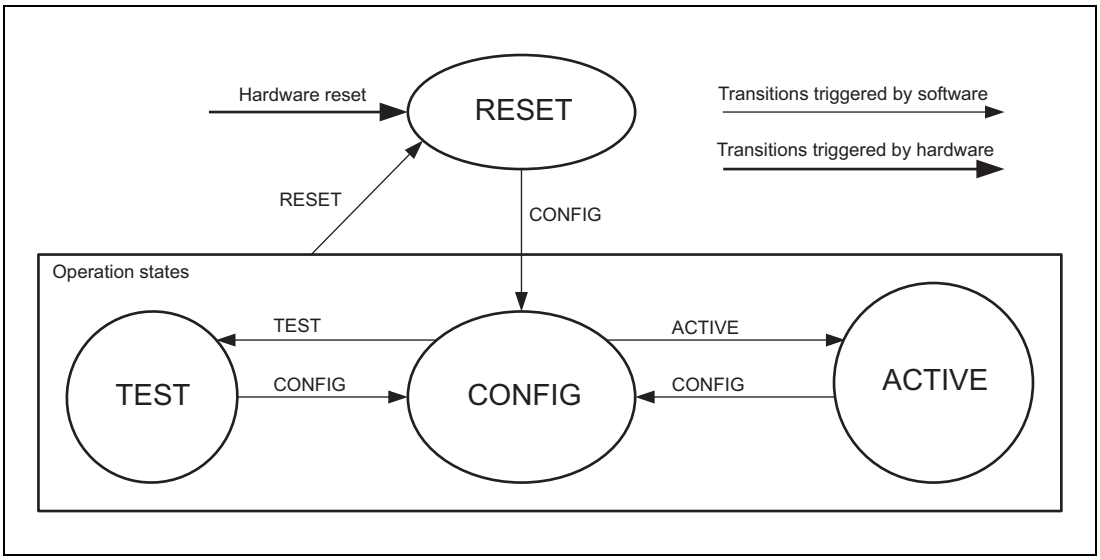


Figure 18.2 Module State Transitions

Figure 18.2 illustrates the implemented operation states and the possible state transitions. The RESET state can be entered from all operation states (TEST, CONFIG, ACTIVE).

18.3.1 Operation State Description

18.3.1.1 RESET State

The RESET state is considered as the power down and reset state of the RHSB module. All internal clocks are stopped to reduce the power consumption. All configuration, control and status registers are reset.

The user can only write to RHSBjGC.RHSBjOPS to leave the RESET state. All other registers are reset and writing to these registers does not have any effect.

18.3.1.2 CONFIG State

Within the CONFIG state the user has write access to all registers to define the module configuration.

18.3.1.3 ACTIVE State

In the ACTIVE state the RHSB module is able to perform up- and downstream communication.

Upstream communication can only be used in ACTIVE state when this function is enabled by RHSBjUCR.RHSBjUE.

Downstream communication can be individually enabled and disabled during ACTIVE and TEST state in the RHSBjDTC register.

18.3.1.4 TEST State

The TEST state is intended to support key-on testing and fault localization. By loop-back the downstream data to the upstream channel, the RHSB internal data paths can be checked.

Refer to **Section 18.4.4, Test Mode Operation** for details and restrictions of this mode.

18.3.2 Activation of the RHSB Module (Leaving RESET State)

To leave the RESET state this flow should be used. The flow assumes that the peripheral bus clock and the RHSB engine clock are already enabled.

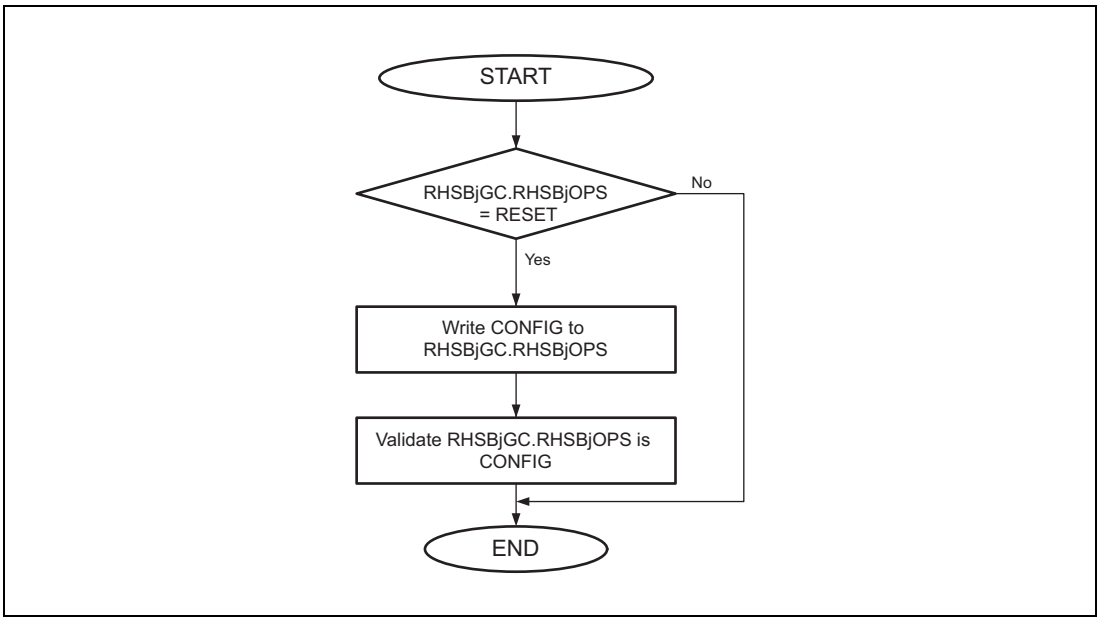


Figure 18.3 Software Flow to Leave RESET State

Note that port function and direction registers of the MCU needs to be configured before the RHSB module can perform active operation. To prevent invalid levels on the downstream chip select lines, it is recommended to define the correct active level for these lines before setting the MCU pin function to the RHSB.

18.3.3 Deactivation of the RHSB Module (Entering RESET State)

To enter the RESET state this flow should be used. Once this flow is executed the peripheral bus clock and the RHSB engine clock can be disabled; the function of the MCU ports can be changed.

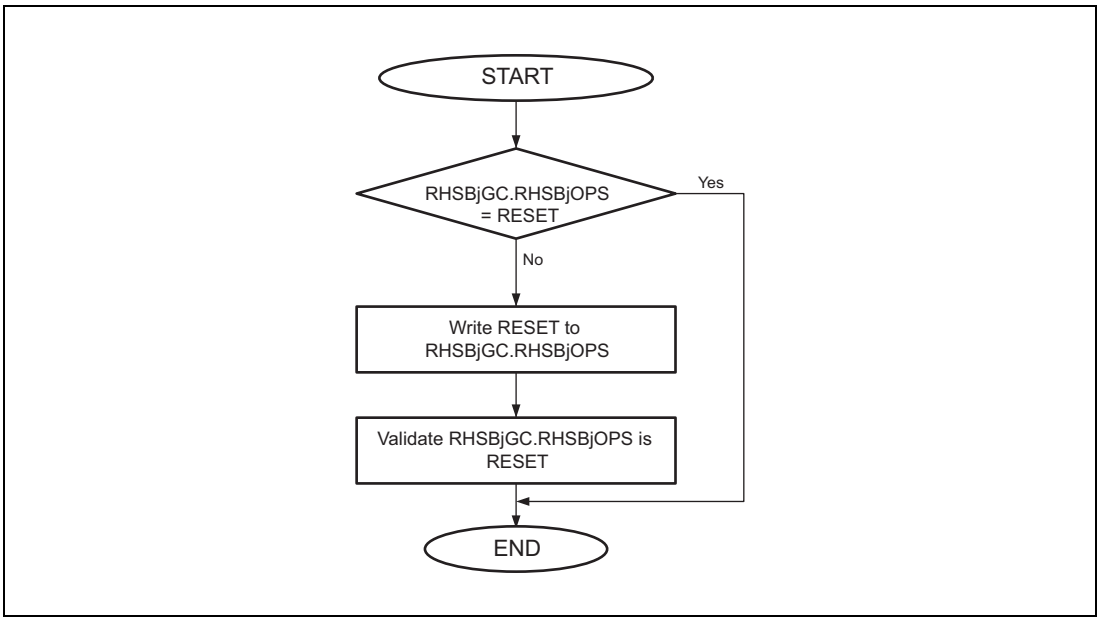


Figure 18.4 Software Flow to Enter RESET State

If the module enters the RESET state, ongoing transmission will be aborted, ongoing reception is lost and the RHSB output lines are changed to their reset levels.

Note when using the flow mentioned in **Figure 18.5**, it is possible to enter RESET state without influencing an ongoing transmission.

18.3.4 Changing between Operation States (ACTIVE, CONFIG, TEST)

Changing the operation state of the RHSB module is triggered by writing the target state to RHSBjGC.RHSBjOPS. The successful processing of the state transition is observable in RHSBjGC.RHSBjOPS.

18.3.5 Leaving ACTIVE State without Interrupting Ongoing Transmission

When the ACTIVE state is left, an ongoing transmission is aborted. This may result in an invalid reception on the downstream receiver side (slave).

Using this flow, interruption of an ongoing transmission can be prevented when changing the operation state to CONFIG or RESET.

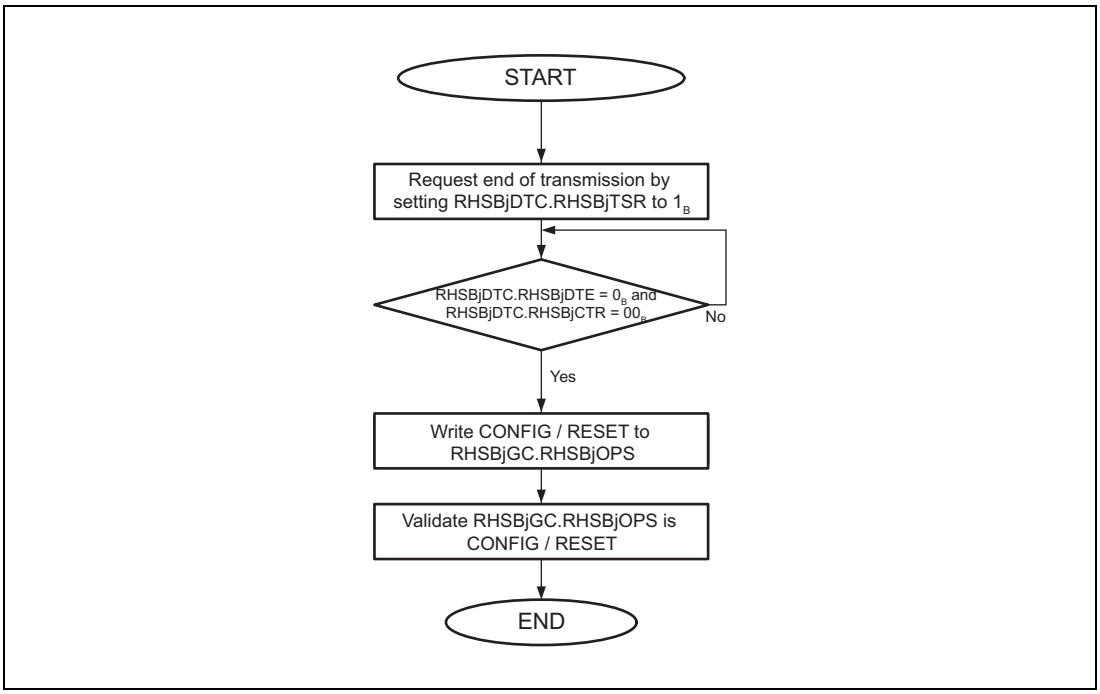


Figure 18.5 Software Flow to Leave ACTIVE/TEST State without Interruption of Ongoing Transmissions

The same flow can be used to leave the TEST state.

18.4 RHSB operations

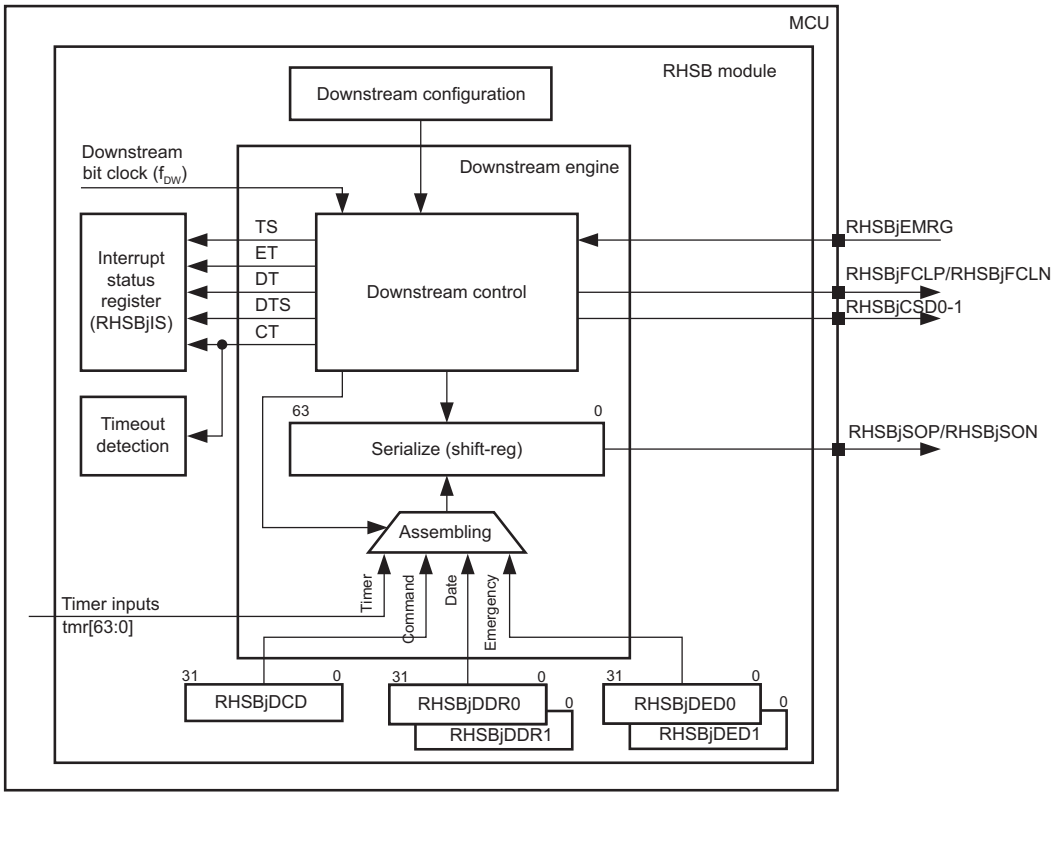
18.4.1 Downstream Communication

The RHSB module uses for downstream communication the synchronous serial transmission.

In addition to the synchronous serial frame, the downstream communication defines a set of communication phases to serve up to two slaves by one frame and to define a repetition period for cyclic transmission.

There are different modes and configuration parameters available to schedule the downstream communication according the application requirements. This downstream configuration is defined by the registers described in **Section 18.2.4, Downstream (Tx) Registers**.

The diagram in **Figure 18.6** illustrates the functional blocks of the downstream engine. The detailed downstream communication is explained in this section.



Note: j = 0,1

Figure 18.6 Downstream Block Diagram

Downstream Configuration

This block represents the downstream related configuration parameters and the downstream control interface handled by the application software.

Downstream Control

This block provides the main function of the downstream engine. It schedules the transmission, controls the frame assembling, generates the downstream communication signals (RHSBjFCLP/ RHSBjFCLN (j = 0, 1), RHSBjCSD0-1 (j = 0, 1)) and triggers the downstream related status flags in the RHSBjIS register.

Serialize

The shift register captures the assembled data at frame start and provides the downstream phase related information to the serial data line (RHSBjSOP/RHSBjSON (j = 0, 1)) of the downstream interface.

Assembling

This multiplexing logic selects the data source for each data bit.

18.4.1.1 Downstream Communication Phases

The downstream communication is composed of different phases.

These phases are used by the RHSB module.

Active phases:

- Assertion phase: The serial data line is invalid
  This optional phase allows the selected slave to get prepared for the content phase.
- Content phase: The serial data line is valid
  During this phase content is transmitted to the selected slave.
- Deassertion phase: The serial data line is invalid
  Optional phase after content is transmitted to the selected slave.

Passive phases:

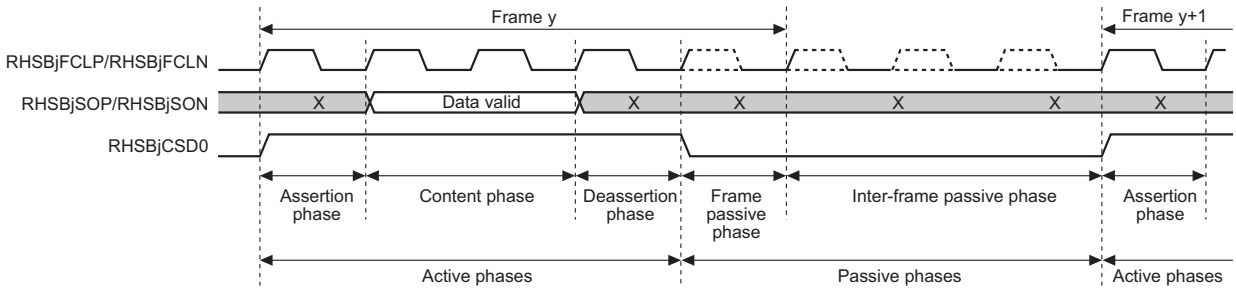
- Frame passive phase:
  This phase is part of each frame and defines a minimum time of inactivation before the next frame starts.
- Inter-frame passive phase:
  This phase describes the time, where no frame is transmitted (bus is idle).
  An inter-frame passive phase occurs, depending on the downstream mode and the downstream configuration.

A frame consists of the active phases and the frame passive phase.

Figure 18.7 illustrates the phases of downstream communication. The frame shown in this figure addresses one slave (slave 0). The mode dependent clock during the passive phases is shown by dotted lines. All phases, including the optional ones, are shown in the ordering used for transmission.

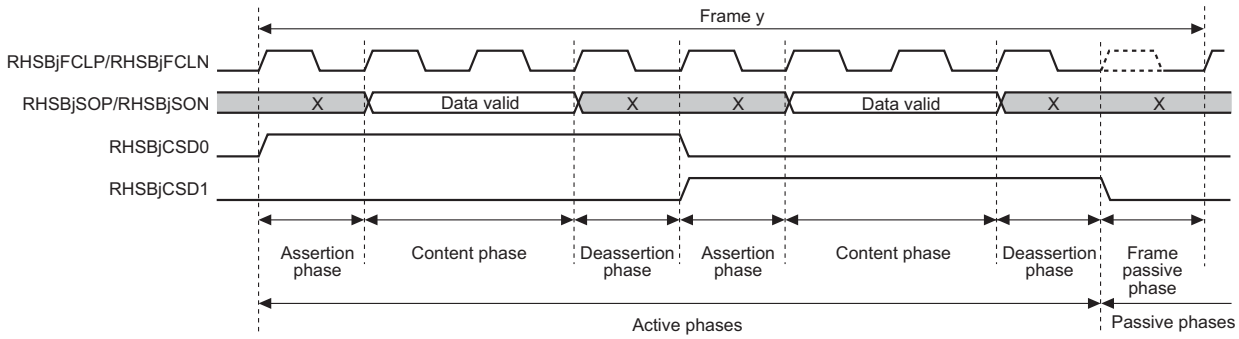
Figure 18.8 illustrates the phases of downstream communication of a frame addressing two slaves (slave 0 and slave 1). The optional assertion and deassertion phases are individually configurable for each slave.

In **Figure 18.7** and **Figure 18.8**, the select signal of the slaves are configured as active high; the clock line phase is set to rising edge.



Note: j = 0,1

Figure 18.7 Downstream Communication Phases



Note: j = 0,1

Figure 18.8 Downstream Communication Phases of a Frame Addressing Two Slaves

For evaluation purpose the user can monitor the communication phases by reading RHSBjMSR. Note that this function is not available for all downstream bit rates.

18.4.1.2 Frame Dependent Flagging

Figure 18.9 illustrates the RHSB internal processing flow of command and data frame transmission. The scheduling when a command or data frame is transmitted is mode dependent and explained in **Section 18.4.1.3, Downstream Modes**.

The flows described in **Figure 18.8** are used as macros in the mode description of **Section 18.4.1.3, Downstream Modes**.

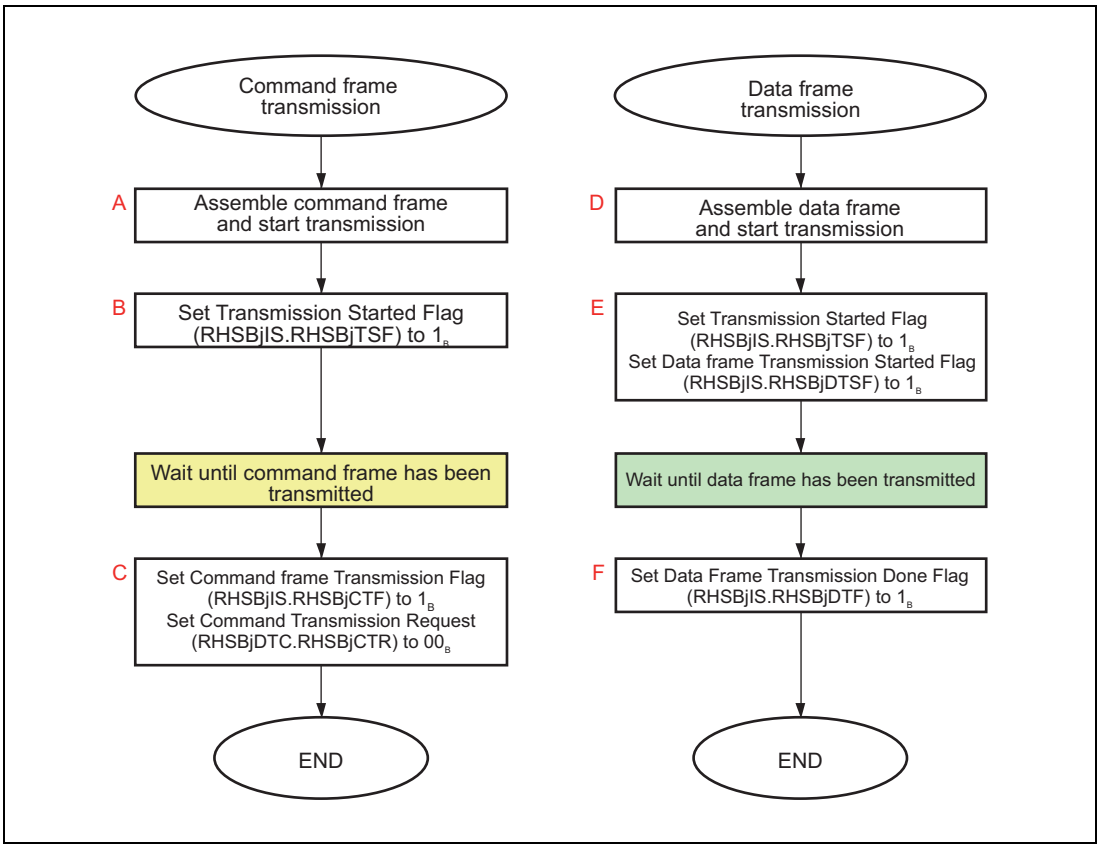


Figure 18.9 Internal Transmission Processing of Command and Data Frames

The flagging related to command and data frame transmission is illustrated in **Figure 18.10**. A gray background illustrates values which depend on previous user interaction but have no influence to the transmission.

In triggered mode RHSBjIS.RHSBjDTSF is always 0_B immediately before a data frame transmission starts. In other modes the value can be 0_B or 1_B.

The timing marks A to F refer to the marks in **Figure 18.9**.

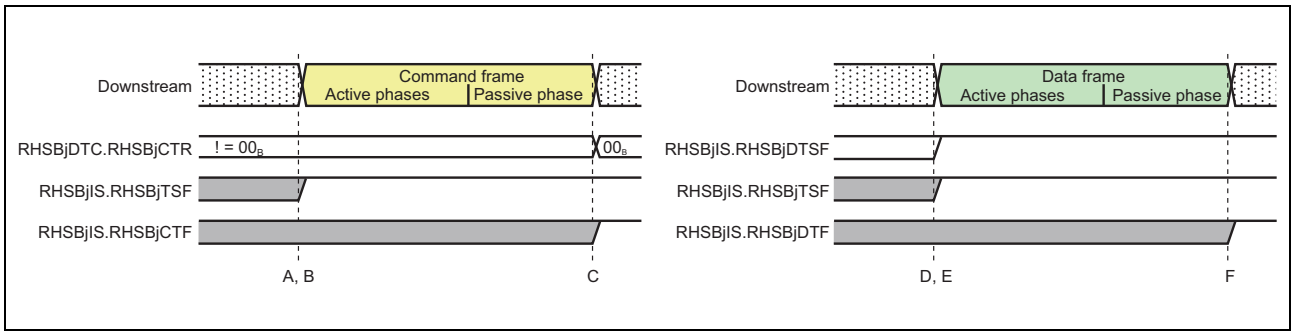


Figure 18.10 Command and Data Frame Related Flagging

When an emergency frame is transmitted, the RHSBjIS.RHSBjETF is rising at the same time as RHSBjIS.RHSBjDTF (timing F).

18.4.1.3 Downstream Modes

The RHSB module supports different downstream transmission modes which can be selected during CONFIG state.

These modes are available and can be configured in RHSBjDCR.RHSBjDMS:

- Triggered mode
  The RHSB module starts downstream communication immediately on user request (software triggered).
- Single-period repetition mode
  The RHSB module starts downstream communication in a defined repetition period. The cyclic transmission is independent from user interaction.
- Multi-period repetition mode
  The RHSB module starts downstream communication in two defined repetition periods. The cyclic transmission is independent from user interaction.

In all modes transmission of data frames is only possible when the data transmission is enabled by setting RHSBjDTC.RHSBjDTE to 1_B.

Except in triggered mode it is recommended to initialize the downstream data registers with valid data before enabling the data transmission because a data frame transmission is immediately started when RHSBjDTC.RHSBjDTE changes from 0_B to 1_B.

The RHSB module is able to transmit command frames independent from the downstream mode and whether data transmission is enabled or not.

(1) Triggered mode

In this mode transmission of data frames is triggered by the user. Each trigger starts the transmission of one data frame.

There is no repetition period defined for this mode, the transmission is controlled by the software execution timing.

These configuration parameters are ignored in triggered mode:

- Command frame insertion mode (RHSBjDCR.RHSBjCIM)
- Command frame transmission delay (RHSBjDCR.RHSBjCTD)
- Repetition period length (RHSBjDCR.RHSBjREP)

A data frame transmission is triggered when RHSBjIS.RHSBjDTSF is 0_B. This bit can be cleared by writing 0_B to RHSBjIS.RHSBjDTSF (keep data unchanged) or by updating the downstream data registers (RHSBjDDRi). Please refer to **Section 18.4.1.10, Data Update and Data Frame Transmission Request** for details about data update and data frame transmission request.

The flow shown in **Figure 18.11** assumes that the RHSB module is in triggered mode and that the data transmission is enabled.

The details about command and data frame transmission (colored ovals) are shown in **Figure 18.9**.

The flow is started when the following configuration is applied:

- RHSBjGC.RHSBjOPS is ACTIVE
- RHSBjDCR.RHSBjDMS is triggered
- RHSBjDTC.RHSBjDTE is 1_B

The flow is immediately aborted when the RHSB module leaves the ACTIVE state.

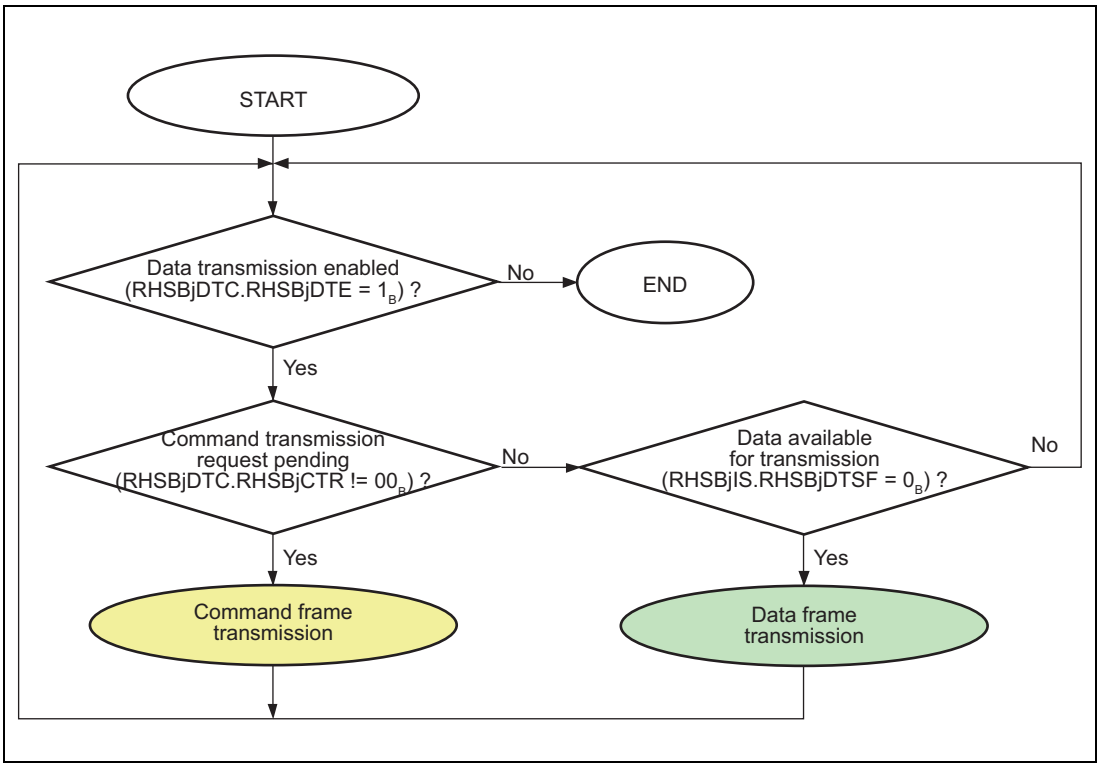


Figure 18.11 Internal Transmission Processing in Triggered Mode when Data Transmission is Enabled

Figure 18.12 illustrates the downstream communication in case of concurrent command and data frame transmission requests. If there is a pending request when a frame transmission ends, the next frame is transmitted immediately.

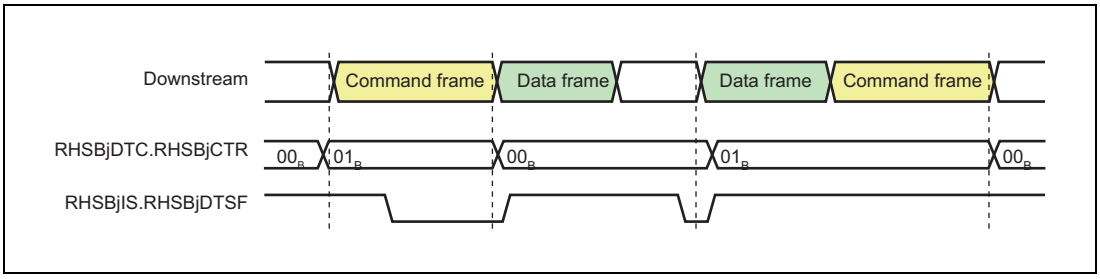


Figure 18.12 Example of Concurrent Command/Data Transmission Requests in Triggered Mode

(a) Command frame transmission in triggered mode

Command frame transmission has higher priority than data frame transmission as shown in the transmission processing figure. The user needs to schedule data and command frame transmission to be consistent with the requirements of the application.

When the data transmission is disabled (RHSBjDTC.RHSBjDTE = 0_B) a command frame is immediately transmitted when the request is pending. This command frame transmission is not controlled by the internal transmission processing as shown in **Figure 18.11**.

(2) Repetition mode (Single-period repetition, Multi-period repetition)

In this mode transmission of data frames is done periodically.

A periodical time tick defines the repetition period of the cyclic data transmission.

The fixed repetition period used in this mode is controlled by the RHSB module.

By RHSBjDTC.RHSBjDTE and RHSBjDTC.RHSBjTSR the user can start and stop the cyclic data transmission in repetition mode. The RHSB module ensures that disabling of data transmission will not interrupt an ongoing transmission.

In repetition mode the trigger of data frame transmission is independent from RHSBjIS.RHSBjDTSF.

However the values from the downstream data registers (RHSBjDDRi) are used to assemble the transmission data.

(a) Repetition period definition

The repetition period is defined by RHSBjDCR.RHSBjREP. The minimum repetition period should be at least the data frame length.

When the repetition period is bigger than the data frame length, there is an inter-frame passive phase between two data frames as illustrated in **Figure 18.13**.

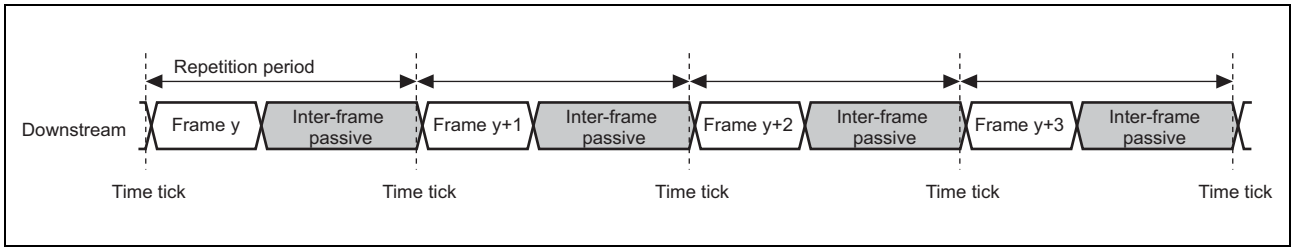


Figure 18.13 Transmission Timing in Repetition Mode

The length of a data frame can be calculated as sum of these configuration parameters:

- Assertion phase length (RHSBjSDCi.RHSBjAPLn) of all addressed slaves
- Data frame selection bit (RHSBjSDCi.RHSBjCPSn) of all addressed slaves
- Deassertion phase length (RHSBjSDCi.RHSBjDPLn) of all addressed slaves
- Number of data bits (RHSBjDEC.RHSBjNDBm) of all used DFTEs (RHSBjDCR.RHSBjSLS and RHSBjDCR1.RHSBj.SLS1 (multi-period repetition mode))
- Data frame passive phase length (RHSBjDCR.RHSBjDFP)

The user should not configure a repetition period (RHSBjDCR.RHSBjREP) less than 8 peripheral bus clock cycles. As example: With a typical peripheral bus clock (CLK_LSB) frequency of 40 MHz, this results in a minimal repetition period of 0.2 µs.

The transmission period count in single-period repetition mode holds “period 0” and that in multi-period repetition mode toggles between “period 0” and “period 1” repeatedly as illustrated in **Figure 18.13**.

The sequence length in single-period repetition mode is defined by RHSBjDCR.RHSBjSLS for period 0. The sequence length in multi-period repetition mode is defiled by RHSBjDCR.RHSBjSLS for period 0 and RHSBjDCR1.RHSBjSLS1 for period 1.

(b) Command frame transmission

Command frame transmission has higher priority than data frame transmission. Depending on the configuration parameter RHSBjDCR.RHSBjCTD transmission of consecutive command frames can be prevented.

Different command frame insertion methods are available. They differ in the strategy for the case when a command frame is transmitted and how data transmission is affected by command frame insertion. All insertion methods (controlled by RHSBjDCR.RHSBjCIM) are explained in detailed in **18.4.1.3 (3) Command frame insertion methods in single-period repetition mode**.

(3) Command frame insertion methods in single-period repetition mode

In Single-Period REPETITION mode, a request for command transmission comes asynchronously with respect to the repetition period. The RHSB module supports different strategies controlled by RHSBjDCR.RHSBjCIM, on how to insert command frames into the repetition schedule.

If data transmission is disabled (RHSBjDTC.RHSBjDTE is 0_B), a command frame is immediately transmitted when the request is pending. In addition the restriction about consecutive command frames (RHSBjDCR.RHSBjCTD is 1_B) is ignored.

(a) Time-slot method

When this method is selected, the RHSB module processes command and data frames aligned to the time ticks. The command frames substitute data frames within the repetition period.

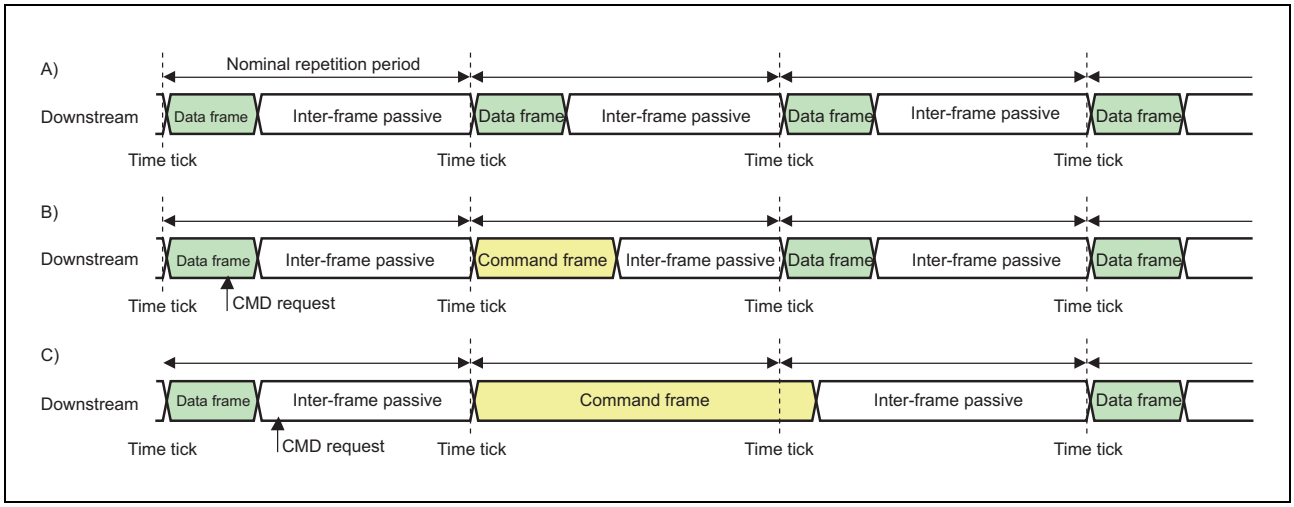


Figure 18.14 Examples of Command Insertion Using Time-Slot Method

Scenario A) shows a schedule with no command requests.

Scenario B) shows command frame insertion where the length of the command frame is less or equal to the repetition period. One data frame is substituted.

Scenario C) shows command frame insertion where the length of the command frame is greater than the repetition period. Two data frames are substituted.

(b) Immediate method

When this method is selected, the RHSB module processes a command frame transmission request as soon as possible (immediately during inter-frame passive phase or when a data frame ends). The command frame transmission is not aligned to the repetition period.

If a data frame transmission cannot be started at the origin time tick because a command frame has broken the repetition period, the data transmission is processed as soon as possible. The repetition period is re-adjusted.

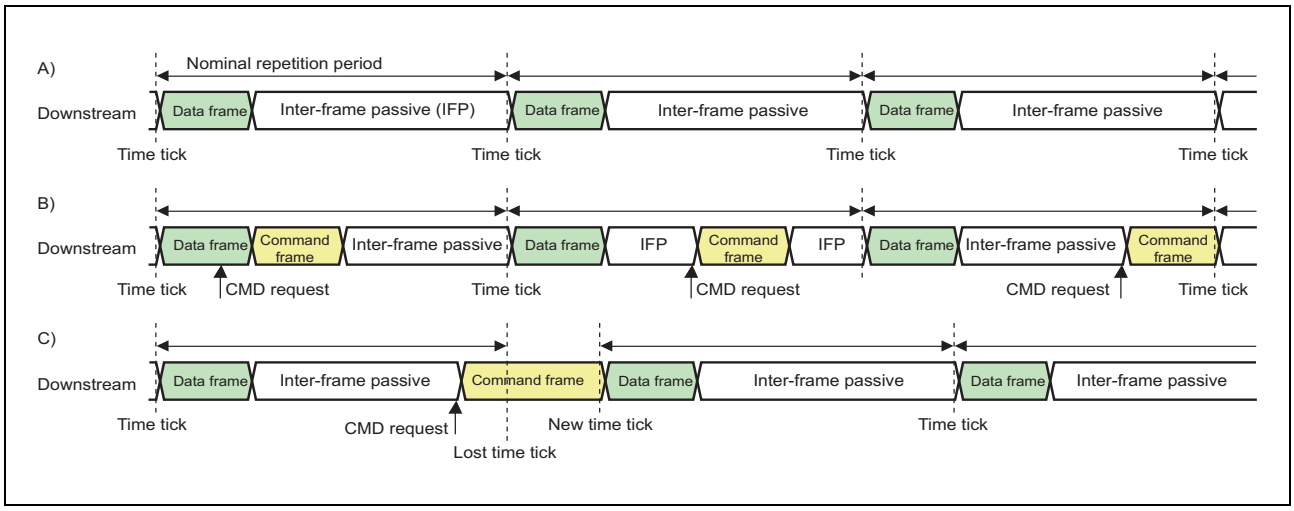


Figure 18.15 Examples of Command Insertion Using Immediate Method

Scenario A) shows a schedule with no command requests.

Scenario B) shows command frame insertion which is not breaking the repetition period (three examples).

Scenario C) shows command frame insertion which is breaking the repetition period. The original time tick for the data frame is lost. When the command frame transmission ends, the time base is re-adjusted, a new time tick is generated and data frame transmission is immediately stated.

It is possible to insert more than one command frame between two data frames if this is allowed from the configuration. Also in this case a re-adjust of the repetition period happens if a command frame brakes the repetition period.

(c) Best-effort method

When this method is selected, the RHSB module starts command frame transmission only

- at the end of a data frame transmission or
- when the command frame transmission will end within the inter-frame passive phase.

The repetition time grid is not adjusted when using the best-effort insertion method.

If the command frame is shorter or equal to the inter-frame passive phase, command frame transmission has no influence to the data frame transmission (see case B in **Figure 18.18**). If possible from configuration, more than one command frame can be transmitted during the repetition period.

If the command frame is longer than the inter-frame passive phase, command frame transmission influences the data frame transmission (see case C in **Figure 18.18**). The data transmission is processed as soon as possible.

If a data frame transmission is influenced by another data frame (shifted by command frame transmission), the data frame is not processed (see case D in **Figure 18.18**).

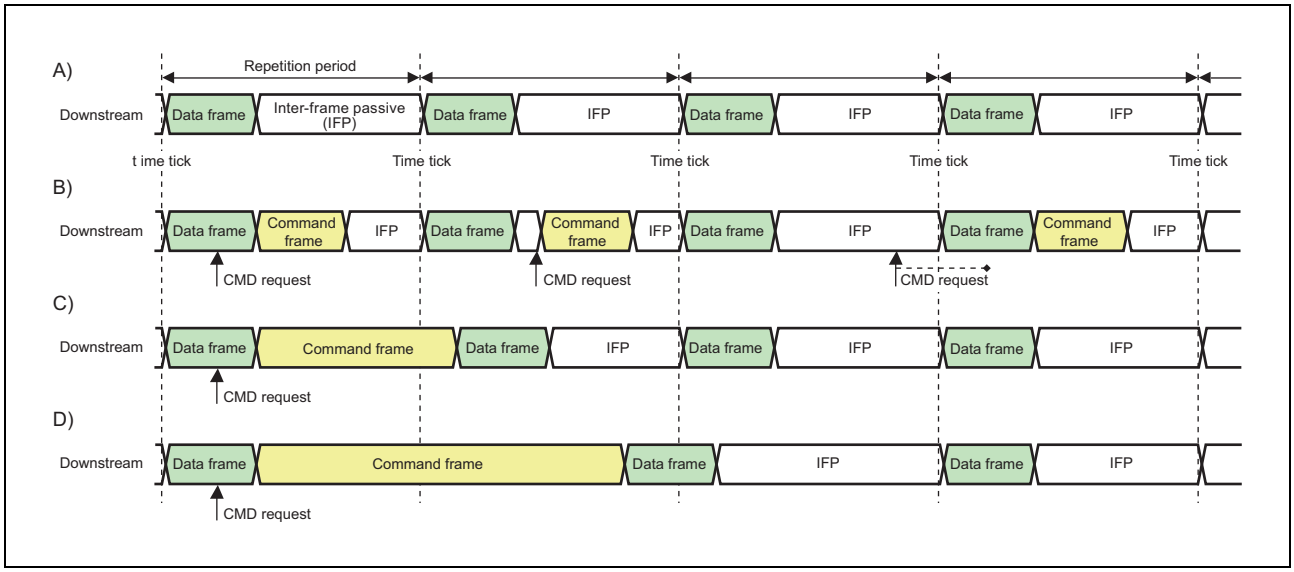


Figure 18.16 Examples of Command Insertion Using Best-Effort Method

Scenario A) shows a schedule with no command requests.

Scenario B) shows command frame insertion where the command frames are shorter or equal to the inter-frame passive length (three examples). In the third example the command frame is shifted after the next data frame, because it would influence the data transmission.

Scenario C) shows command frame insertion where the command frames are longer than the inter-frame passive length. The delayed data frame has no influence to the next data frame.

Scenario D) shows command frame insertion where the command frames are longer than the inter-frame passive length. The delayed data frame influences to the next data frame.

18.4.1.4 Command Frame Insertion Methods in Multi-Period Repetition Mode

In the Multi-Period repetition mode, a request for command transmission comes asynchronously with respect to the repetition period.

(1) Best-effort method

Multi-Period repetition mode supports only Best-effort method. When this method is selected, the RHSB module starts command frame transmission only

- at the end of a data frame transmission in period 1.

The repetition time grid is not adjusted when using the best-effort insertion method.

If the command frame is shorter or equal to the inter-frame passive phase in period 1, command frame transmission has no influence to the data frame transmission (see case B, C, D, E, F and G in **Figure 18.17**). If possible from configuration, more than one command frame can be transmitted during the repetition period.

If the command frame is longer than the inter-frame passive phase, command frame transmission influences the data frame transmission (see case H in **Figure 18.17**). The transferred command and data are not guaranteed.

If a data frame transmission is influenced by another data frame (shifted by command frame transmission), the data frame is not processed (see case I in **Figure 18.17**). The transferred command and data are not guaranteed.

The processing of the best-effort method is also mentioned in transmission processing flows as shown in **Figure 18.17**.

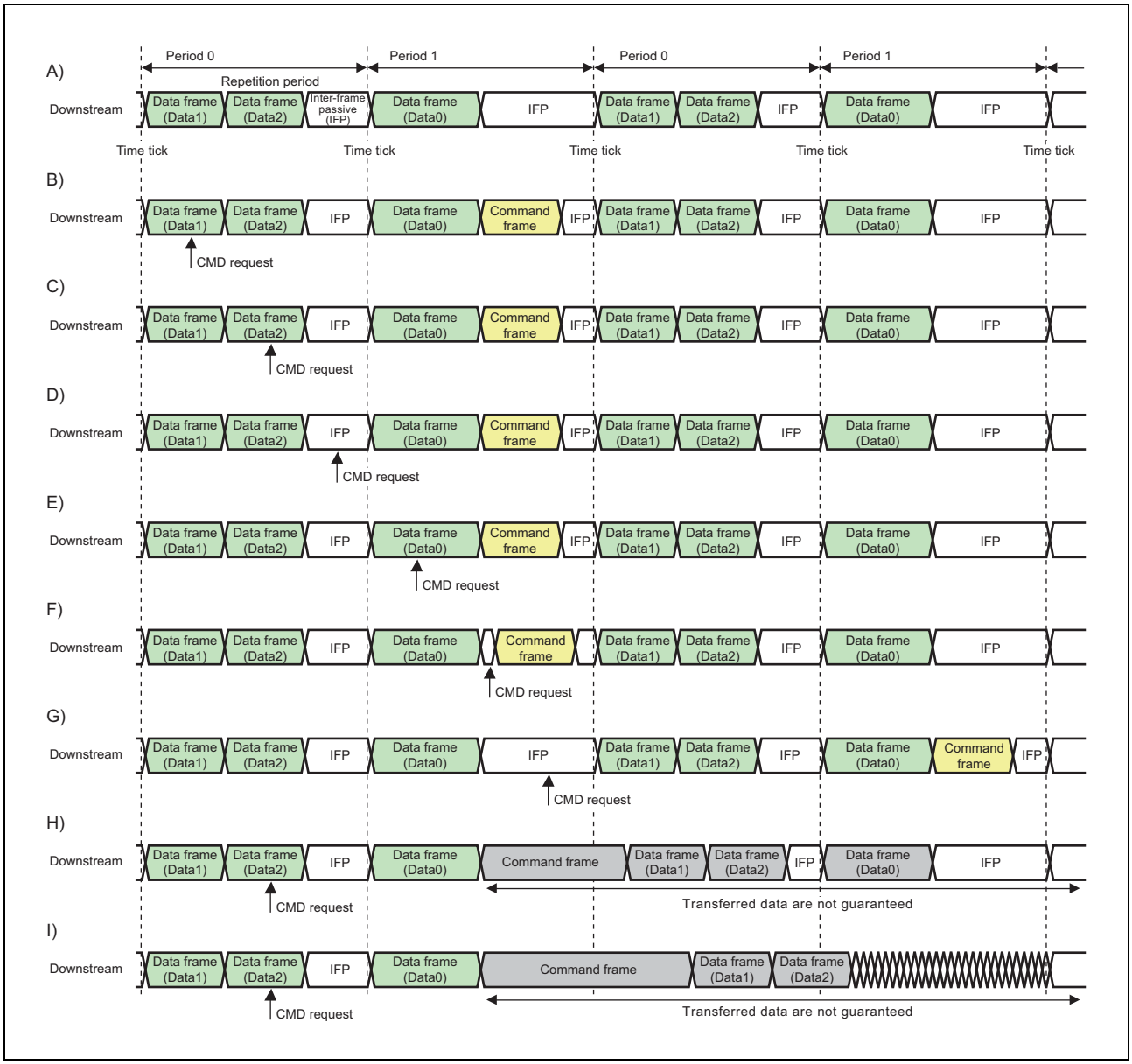


Figure 18.17 Examples of Command Insertion Using Best-Effort Method in Multi-Period Repetition Mode

Scenario A) shows a schedule with no command requests.

Scenario B, C, D) shows command frame insertion when a command request comes in period 0. The command frame is shifted after the end of data frame in period 1.

Scenario E) shows command frame insertion when a command request comes at a data frame in period 1. The command frame is shifted after the end of data frame in period 1.

Scenario F) shows command frame insertion when a command request comes at an inter-frame passive phase and the command length is shorter than the rest of the inter-frame passive length.

Scenario G) shows command frame insertion when a command request comes at an inter-frame passive phase and the command length is longer than the rest of the inter-frame passive length.

Scenarios H) and I) show command frame insertion where the command frame length is longer than the inter-frame passive length. This setting is prohibited. Transferred data after the command frame are not guaranteed until the user writes RHSBjDTC.RHSBjTSR to 1_B.

18.4.1.5 Physical Frame Format

(1) Clock line

The serial clock line (RHSBjFCLP/RHSBjFCLN (j = 0, 1)) provides the timing information from the MSC master to the MSC slave modules. The clock line and the clock phase are common for all slaves.

The user can configure by setting RHSBjDCR.RHSBjCLP whether the serial data line (RHSBjSOP/RHSBjSON (j = 0, 1)) and the chip select lines (RHSBjCSD0-1 (j = 0, 1)) are changing their values with rising edge or falling edge of the serial clock line (RHSBjFCLP/RHSBjFCLN (j = 0, 1)).

Note that RHSBjDCR.RHSBjCLP is directly influencing the level on the serial clock line (RHSBjFCLP/RHSBjFCLN (j = 0, 1)).

The user is able to configure the serial clock activity by RHSBjDCR.RHSBjCAC anytime. The RHSB module guarantees a glitch free clock signal on the serial clock line (RHSBjFCLP/RHSBjFCLN (j = 0, 1)).

The configuration given by RHSBjDCR.RHSBjCAC is not affecting the clock activity during the active phases; for a proper downstream communication the clock is required during active phases. RHSBjDCR.RHSBjCAC is only affecting the clock activity during the passive phases; for downstream communication the clock is not required during passive phases.

An active serial clock line may be required in some applications by a connected slave (e.g. if a slave uses the downstream clock to handle the upstream communication).

(2) Individual slave configuration

For each slave the user can independently program in CONFIG state:

- The active level of the chip select line (RHSBjSDCi.RHSBjCSLPn)
- The active level of the serial data line (RHSBjSDCi.RHSBjSOLPn)
- The assertion phase length (RHSBjSDCi.RHSBjAPLn)
- The deassertion phase length (RHSBjSDCi.RHSBjDPLn)
- The presence of a selection bit in data frames (RHSBjSDCi.RHSBjCPSn)

Note that the active level configuration for the chip select lines is directly visible at the outputs of the RHSB module.

(3) Downstream frame encoding

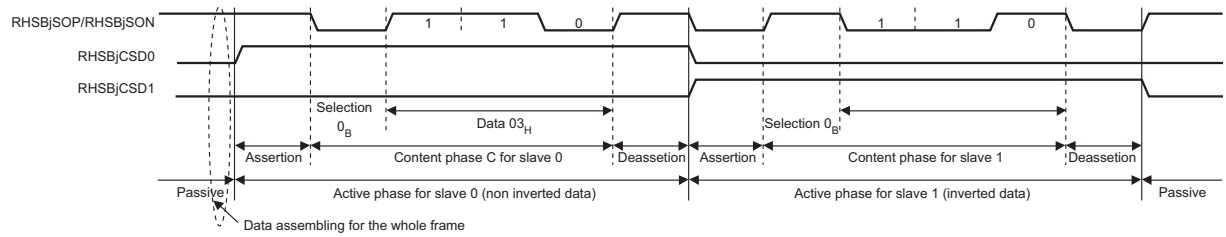
The selection bit is 0_B for data frames and 1_B for command frames.

The serial data line (RHSBjSOP/RHSBjSON (j = 0, 1)) is specified as 'invalid' for the assertion and deassertion phases; the RHSB module is transmitting 1_B during these phases. Note that the level on the serial data line depends on the individual slave configuration (RHSBjSDCi.RHSBjSOLPn).

An enabled inversion of the serial data line happens as long as slave n is selected.

The inversion of the serial data line is only configurable for active phases; the RHSB module sets serial data line (RHSBjSOP/RHSBjSON (j = 0, 1)) to high level during passive phases independent from RHSBjSDCi.RHSBjSOLPn.

Figure 18.18 illustrates a data frame encoding in case of two slaves with active high chip select lines. The frame transmits a data word of 3 bits (value is 03_H) to slave 0 and to slave 1. The configuration of both slaves is identical except that the data line for the slave 1 is inverted.



Note: j = 0,1

Figure 18.18 Frame Encoding Example

The RHSB module assembles the downstream frame directly before the transmission starts (as marked in **Figure 18.18**). This assembling timing is independent from the downstream mode (RHSBjDCR.RHSBjDMS) and the frame type (command/data frame).

18.4.1.6 Data Frame Assembling

A data frame is a stream of un-correlated bits sent to up to 4 slaves. There is an individual chip select line for each slave indicating which content is for which slave. Two chip select signals are not simultaneously active. Only one chip select signal can be activated at a time.

A data frame is composed by up to 4 Downstream Frame Transmission Elements (DFTEs).

The number of DFTEs used is defined by RHSBjDCR.RHSBjSLS.

For each DFTE the user can configure:

- The slave assigned to (RHSBjDEC.RHSBjSSDm)
- Presence of an optional selection bit in each content phase (RHSBjSDCi.RHSBjCPSn)
- The number of used data bits (RHSBjDEC.RHSBjNDBm)

If a slave needs more than one DFTE, the user should assign consecutive DFTEs to the same slave. These DFTEs are combined into one content phase.

Note that it is possible to assign nonconsecutive DFTEs to the same slave (e.g. DFTE0 and DFTE2); in this case each DFTE defines its own content phase.

The DFTEs are always placed in ascending order inside a data frame. Because the DFTE/slave assignment is free configurable, the ordering of slave selection is not restricted by the fixed DFTE order.

(1) Data source configuration

For each bit of a DFTE the user can configure independently the data source by setting RHSBjDEBAm.RHSBjDSSn as:

- Timer input (tmr)
- Timer input inverted
- Downstream data register

The relation between DFTEs and the data sources is shown in **Figure 18.29**. The diagram in **Figure 18.19** illustrates this relation for one bit and includes the emergency case.

For details about the mapping of timer inputs please refer to **Section 18.7, Cross Bar (XBAR)**.

Table 18.26 Relation of DFTEs and Data Sources

	Timer Input	Data Input	Bit Assignment	Emergency Data	Emergency Enable
DFTE0[15:0]	tmr[15:0]	DDR0[31:16]	DEBA0[31:0]	DED0[31:16]	EBE0[31:16]
DFTE1[15:0]	tmr[31:16]	DDR0[15:0]	DEBA1[31:0]	DED0[15:0]	EBE0[15:0]
DFTE2[15:0]	tmr[47:32]	DDR1[31:16]	DEBA2[31:0]	DED1[31:16]	EBE1[31:16]
DFTE3[15:0]	tmr[63:48]	DDR1[15:0]	DEBA3[31:0]	DED1[15:0]	EBE1[15:0]

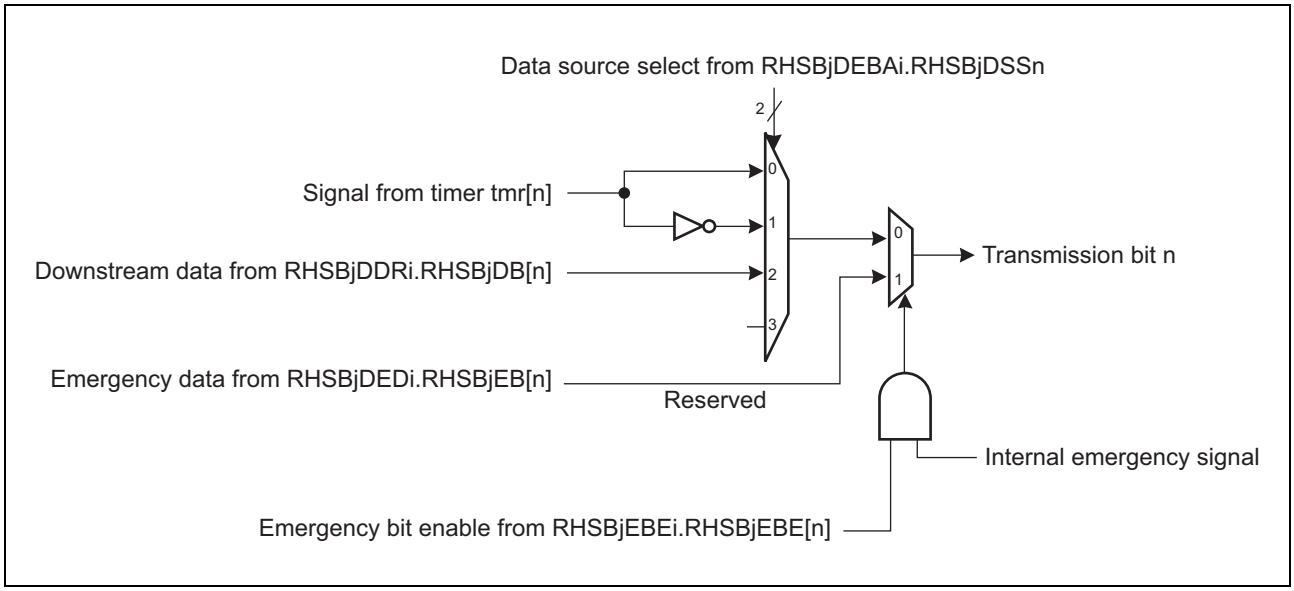


Figure 18.19 Data Source Selection for One Bit of a DFTE

The 'transmission bit n' signal in **Figure 18.19** represents the value of bit n. This signal is used by the RHSB module to assemble the content of a data frame when starting the data frame transmission.

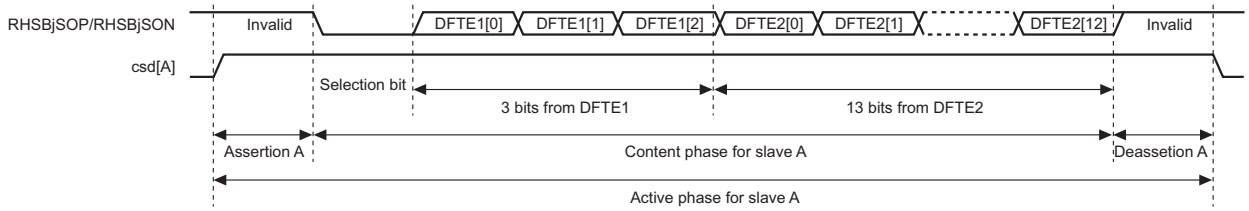
(2) Data frame DFTE mapping

Table 18.27 shows all possible configurations of data bits available for each slave depending on the number of used slaves. The slaves are named A to D to illustrate that there is no dependency between the DFTEs and slave number. If there is a range of assigned DFTEs given, these DFTEs have to be consecutive.

Table 18.27 Combinations of DFTEs and Data Bits Assignments to Slaves

Number of Slaves	Slave A Bits	Slave B Bits	Slave C Bits	Slave D Bits	DFTE Assignment
1 slave (A)	1 to 64	—	—	—	1 to 4 DFTEs to slave A
2 slaves (A, B)	1 to 48	1 to 16	—	—	1 to 3 DFTEs to slave A 1 DFTE to slave B
2 slaves (A, B)	1 to 32	1 to 32	—	—	1 to 2 DFTEs to slave A 1 to 2 DFTEs to slave B

Figure 18.20 shows an example of a downstream frame to be transmitted to slave A assembled from DFTE1 and DFTE2. The slave device A is configured in RHSBjSDCi register as: RHSBjAPLn = 1, RHSBjCSLPn = 0, RHSBjSOLPn = 0, RHSBjCPSn = 1, RHSBjDPLn = 1 (active high, assertion and deassertion phase of 1_D bit, selection bit is present). For DFTE1 3 data bits are configured (RHSBjDEC.RHSBjNDB1 = 2), for DFTE2 13 data bits are configures (RHSBjDEC.RHSBjNDB2 = 12).



Note: j = 0,1

Figure 18.20 Example Bit Mapping when Combining Two DFTEs

18.4.1.7 Command Frame Assembling

The command frame is a stream of correlated bits sent to one slave. There is only one content phase available in a command frame.

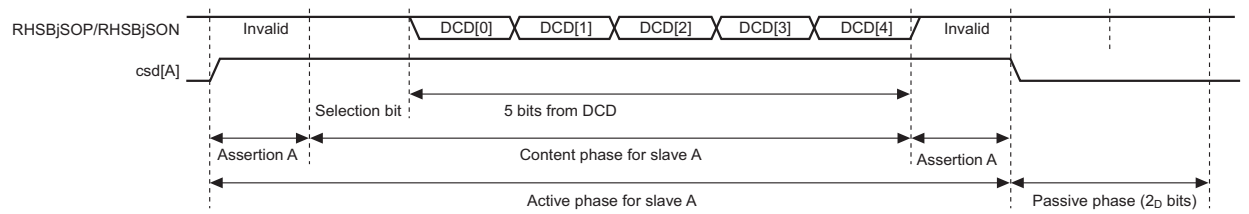
The content phase of a command frame always contains a selection bit.

The number of data bits which are sent in the content phase from the RHSBjDCD register is configurable from 1 to 32 bits. This number can be changed individually for each command frame transmission by RHSBjDTC.RHSBjNCB.

The frame passive phase of command frames is fixed to 2_D bits.

The length of assertion and deassertion phase depends on the slave device configuration in RHSBjSDCi register as used for data frames.

Figure 18.21 shows an example of a command frame with 5 bits (RHSBjDTC.RHSBjNCB = 4) to be transmitted to slave A. The slave device A is configured in RHSBjSDCi register as: RHSBjAPLn = 1, RHSBjCSLPn = 0, RHSBjSOLPn = 0, RHSBjDPLn = 1 (active high, assertion and deassertion phase of 1_D bit).



Note: j = 0,1

Figure 18.21 Command Frame Bit Mapping

18.4.1.8 Emergency Function

The emergency function of the RHSB modules supports an automatic transmission of pre-defined data to slaves in case of emergency. The emergency function controls only the content of data frames; the scheduling of data transmission is not affected.

The external emergency signal (RHSBjEMRG (j = 0, 1)) which is available as primary input pin of the MCU, is evaluated to derive the emergency condition.

The emergency function can be globally enabled/disabled using RHSBjDCR.RHSBjEE. Additionally, the user can individually select via RHSBjEBEi.RHSBjEBE for each bit of each DFTE if there should any action taken in an emergency case.

When the RHSB module detects the emergency condition, the data source of the selected DFTE bits is forced to the corresponding value defined in the downstream emergency data registers (RHSBjDEDi). This selection is illustrated in **Figure 18.19**.

To use the emergency function, the user should:

- configure the active level of the emergency signal (RHSBjDCR.RHSBjEIP),
- configure the mode indicating how the emergency condition is derived from the emergency signal (RHSBjDCR.RHSBjEIM),
- enable the emergency function and define the usage of automatic stop capability (RHSBjDCR.RHSBjEE),
- select the action in an emergency case for the individual DFTE bits (RHSBjEBEi.RHSBjEBE),
- define the emergency data for the individual DFTE bits (RHSBjDEDi.RHSBjEB).

Figure 18.22 illustrates how the emergency condition (internal emergency signal) is derived from the level of the external emergency signal (RHSBjEMRG (j = 0, 1)). The RHSB module requires stable levels on the external emergency signal which are longer than three peripheral bus clock cycles (CLK_LSB).

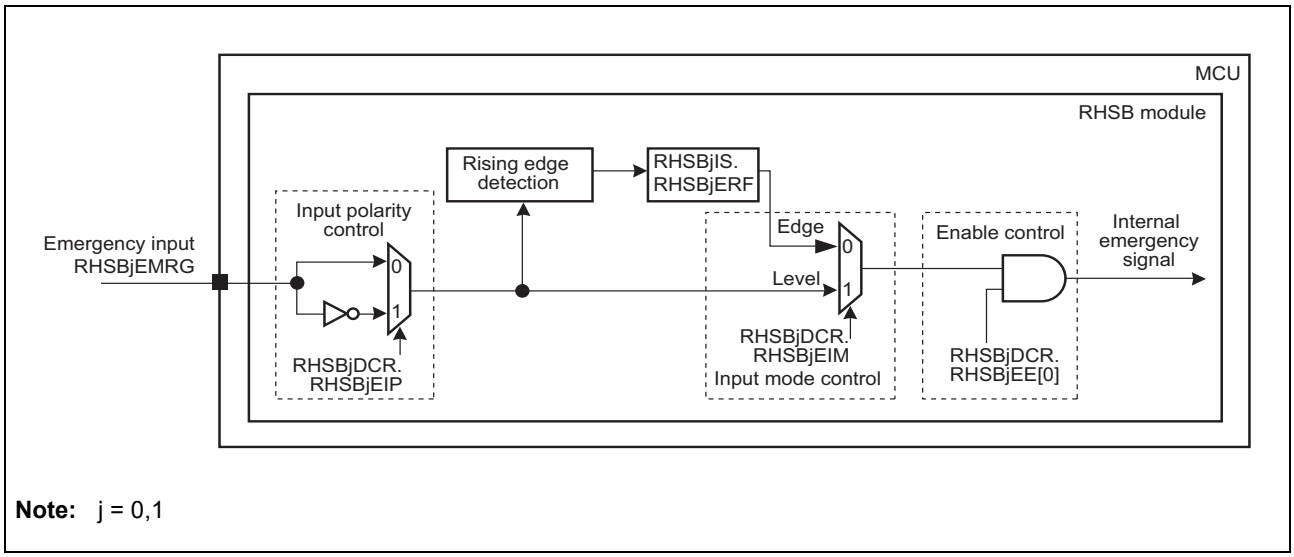


Figure 18.22 Detection of Emergency Condition

The RHSB module uses the internal emergency signal when assembling a data frame. To transmit an emergency frame the internal emergency signal has to be active at assembling time (when the data frame starts).

Even if the emergency function is not used (RHSBjDCR.RHSBjEE is 00_B), the user can read RHSBjIS.RHSBjERF to get informed about the emergency case detection.

(1) Edge sensitive input mode

The emergency input mode is edge sensitive when RHSBjDCR.RHSBjEIM is 0_B

When the RHSB module detects an edge where the external emergency signal goes active, the RHSBjIS.RHSBjERF flag is set to 1_B.

This flag needs to be cleared by the user.

When using the edge sensitive input mode, the user has to ensure that the RHSBjIS.RHSBjERF flag is 1_B until an emergency frame is transmitted. The RHSBjIS.RHSBjETF flag indicates that the emergency frame has been transmitted.

Note that an edge will be detected when the ACTIVE state is entered while there is an active level at the external emergency signal.

(2) Level sensitive input mode

The emergency input mode is level sensitive when RHSBjDCR.RHSBjEIM is 1_B.

When using the level sensitive input mode, the user has to ensure that the external emergency signal is on the active level for one repetition period or longer in single-period repetition mode or for two repetition periods or longer in multi-period repetition mode. If shorter, the RHSB module may not detect emergency.

(3) Automatic stop function

When the emergency function is enabled by RHSBjDCR.RHSBjEE, the user defines if the data transmission should be automatically stopped after the transmission of the first emergency frame.

When RHSBjDCR.RHSBjEE is set to 01_B (emergency condition detection enabled, no stop), the automatic stop function is inactive. The RHSB module transmits emergency frames as long as the emergency condition (see **Figure 18.22**) is true.

When the emergency condition changes to false, the RHSB module continues with transmission of normal data frames.

When RHSBjDCR.RHSBjEE is set to 11_B (emergency condition detection and automatic stop enabled), the RHSB module transmits exactly one emergency frame in single-period repetition mode or the RHSB module transmits one or two emergency frames in multi-period repetition mode when the emergency condition is true.

After transmitting the emergency frame to all slaves, the data transmission is automatically stopped (RHSBjDTC.RHSBjDTE changes to 0_B) after period 0 as illustrated in **Figure 18.23**. Independent from the downstream mode (RHSBjDCR.RHSBjDMS) no data transmission is possible until the user enables this by setting DTC.DTE to 1_B. After setting DTC.DTE to 1_B, the RHSB module restarts transmission of DFTE0 data from period 0.

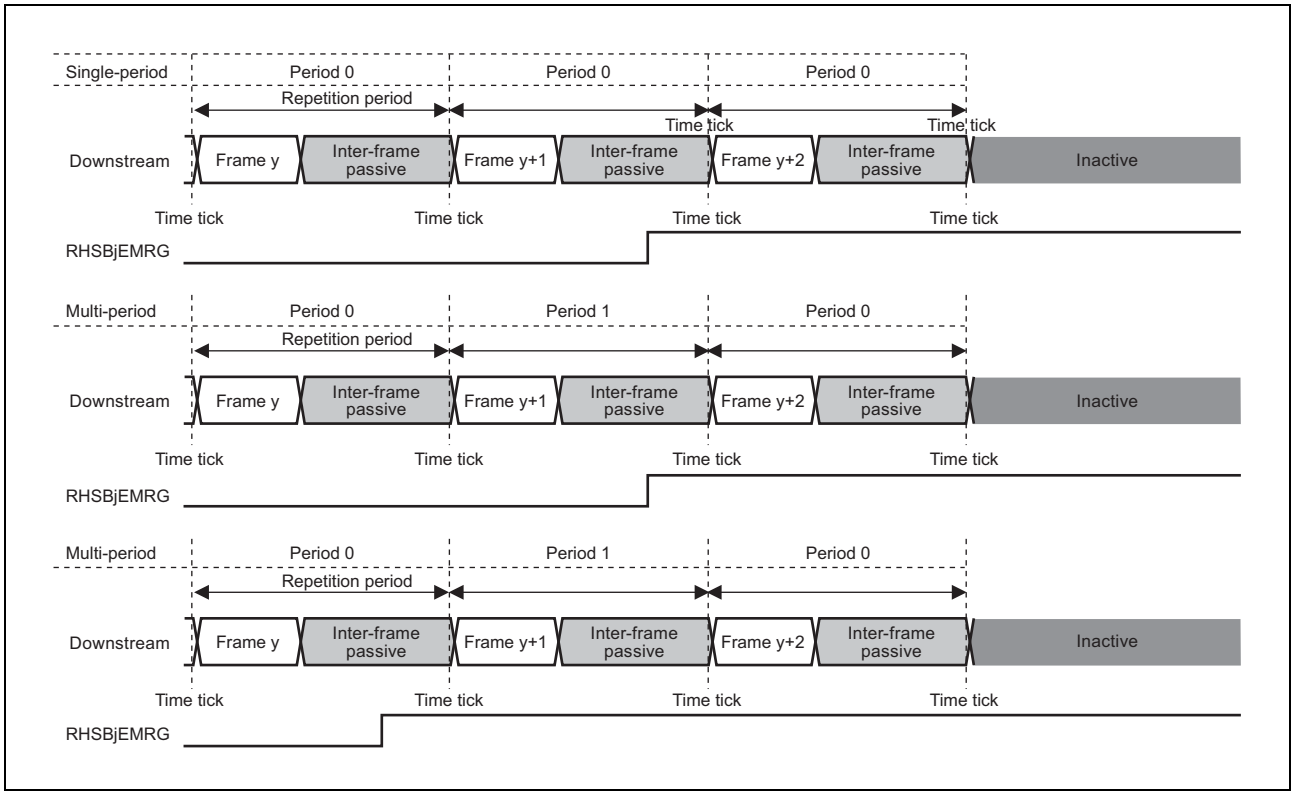


Figure 18.23 Automatic Stop Function in Repetition Mode

18.4.1.9 Downstream Bit Rates

The downstream bit rate (f_{DW}) is derived from the RHSB engine clock source frequency (f_{PE}).

With the divider value x defined by RHSBjDCR.RHSBjDBR, the downstream bit rate can be calculated as function of the RHSB engine clock source frequency:

$f_{DW} = f_{PE} / x$

With f_{PE} of 80 MHz these downstream bit rates can be achieved.

Table 18.28 Possible Downstream Bit Rates with a RHSB Engine Clock Source of 80 MHz

RHSBjDCR.RHSBjDBR	1	3	4	7	8	9	10	11	12
x	2	4	5	8	16	32	64	128	256
Bit rate	40 Mbit/s	20 Mbit/s	16 Mbit/s	10 Mbit/s	5 Mbit/s	2.5 Mbit/s	1.25 Mbit/s	625 Kbit/s	312.5 Kbit/s

When the divider value x is odd, the duty cycle of the clock signals on the serial clock line (RHSBjFCLP/RHSBjFCLN) will not be 50%. In this case, the first phase of the clock signal becomes 1 clock cycle longer than that of f_{PE}.

Example: When x is 5 and RHSBjDCR.RHSBjCLP is 0_B, the widths at high and low level of the clock signal are 3 cycles of f_{PE} and 2 cycles of f_{PE}, respectively. When x is 5 and RHSBjDCR.RHSBjCLP is 1_B, the widths at low and high level of the clock signal are 3 cycles of f_{PE} and 2 cycles of f_{PE}, respectively.

18.4.1.10 Data Update and Data Frame Transmission Request

By setting RHSBjIS.RHSBjDTSF to 0_B, the user acknowledges the data transmission start event and informs the RHSB module that new data is available.

RHSBjIS.RHSBjDTSF can be set to 0_B by two strategies:

- Writing the RHSBjIS.RHSBjDTSF flag directly to 0_B
  - Data configured in RHSBjDDRi is used for the next transmission
- Writing to RHSBjDDRi register (by the user or by DMA)
  - New data written to RHSBjDDRi is used for the next transmission

The write DDRi method allows an optimized transmission flow because it automatically sets RHSBjIS.RHSBjDTSF to 0_B and hence starts the transmission.

It depends on the configuration parameter RHSBjDCR.RHSBjSLS (number of used DFTEs) when a write to a RHSBjDDRi triggers a transmission.

- RHSBjDCR.RHSBjSLS is 00_B (1 DFTE is used): writing to RHSBjDDR0[31:16] (DFTE0)
- RHSBjDCR.RHSBjSLS is 01_B (2 DFTEs are used): writing to RHSBjDDR0[15:0] (DFTE1)
- RHSBjDCR.RHSBjSLS is 10_B (3 DFTEs are used): writing to RHSBjDDR1[31:16] (DFTE2)
- RHSBjDCR.RHSBjSLS is 11_B (4 DFTEs are used): writing to RHSBjDDR1[15:0] (DFTE3)

Even if DCR1.SLS1(number of used DFTEs for period 1) is available in multi-period repetition mode, it depends on the configuration parameter DCR.SLS (number of used DFTEs) when a write to a DDRi triggers a transmission. If DCR1.SLS1 is smaller than DCR.SLS, the user should write dummy data to DDRi defined by DCR.SLS.

- DCR.SLS is 00_B (1 DFTEs is used): writing to DDR0[31:16] (DFTE0)
- DCR.SLS is 01_B (2 DFTEs are used): writing to DDR0[15:0] (DFTE1)
- DCR.SLS is 10_B (3 DFTEs are used): writing to DDR1[31:16] (DFTE2)
- DCR.SLS is 11_B (4 DFTEs are used): writing to DDR1[15:0] (DFTE3)

The data frame transmission started flag (RHSBjIS.RHSBjDTSF) is set to 1_B when the assembling has been performed at the start of the data frame transmission.

RHSBjIS.RHSBjDTSF equal to 1_B indicates that a data frame transmission has been started and so the downstream data registers can be written with new transmit data.

In addition the RHSBjIS.RHSBjDTSF is set to 1_B when data transmission is initially enabled by RHSBjDTC.RHSBjDTE to prevent a data frame transmission without explicit software trigger.

Updating RHSBjDDRi while a data frame is transmitted is not affecting the ongoing transmission but it requests an additional data transmission.

As there is write protection of the downstream data registers (RHSBjDDRi) while there is new data pending for transmission, it is guaranteed, that written downstream data is transmitted at least one time.

(1) Scheduling of Downstream Data Update

The user can update the downstream data by using the following as a trigger.

- Data Frame Transmission Started
  New data can be supplied timely. In this case, RHSBjIS.RHSBjDTSF is used as a trigger.

- Data Frame Transmission Done
  New data is supplied upon completion of data frame transmission. In this case, RHSBjIS.RHSBjDTF is used as a trigger.

Writing to the RHSBjDDRi, the RHSBjIS.RHSBjDTSF and RHSBjIS.RHSBjDTF are cleared. It is recommendable to use only one of them as a trigger of downstream data update.

18.4.1.11 Command Frame Transmission Request

The transmission of command frames is always triggered when the user writes to RHSBjDTC.RHSBjCTR. On each request the transmission of one command frame is started. The method for how command frames are inserted into the downstream communication depends on the selected mode and a set of configuration parameters; for details refer to the detailed mode description.

When RHSBjDTC.RHSBjCTR is set to a value not equal to 00_B, a command frame transmission is scheduled. After completion of the command frame transmission, RHSBjDTC.RHSBjCTR is automatically set to 00_B and the user is able to set a new request.

18.4.2 Upstream Communication

The RHSB module uses for upstream communication the asynchronous serial reception.

There are two upstream channels available to connect up to two slaves with the RHSB module. Only one channel can be selected as active channel at the same time. Any activity on the not selected channels is ignored.

It is possible to receive consecutive upstream frames sent on the same serial input line.

Reception of upstream communication is only supported when the user has enabled this function by setting RHSBjUCR.RHSBjUE to 1_B in CONFIG state.

The diagram in **Figure 18.24** illustrates the functional blocks of the upstream engine. The detailed upstream communication is explained in this section.

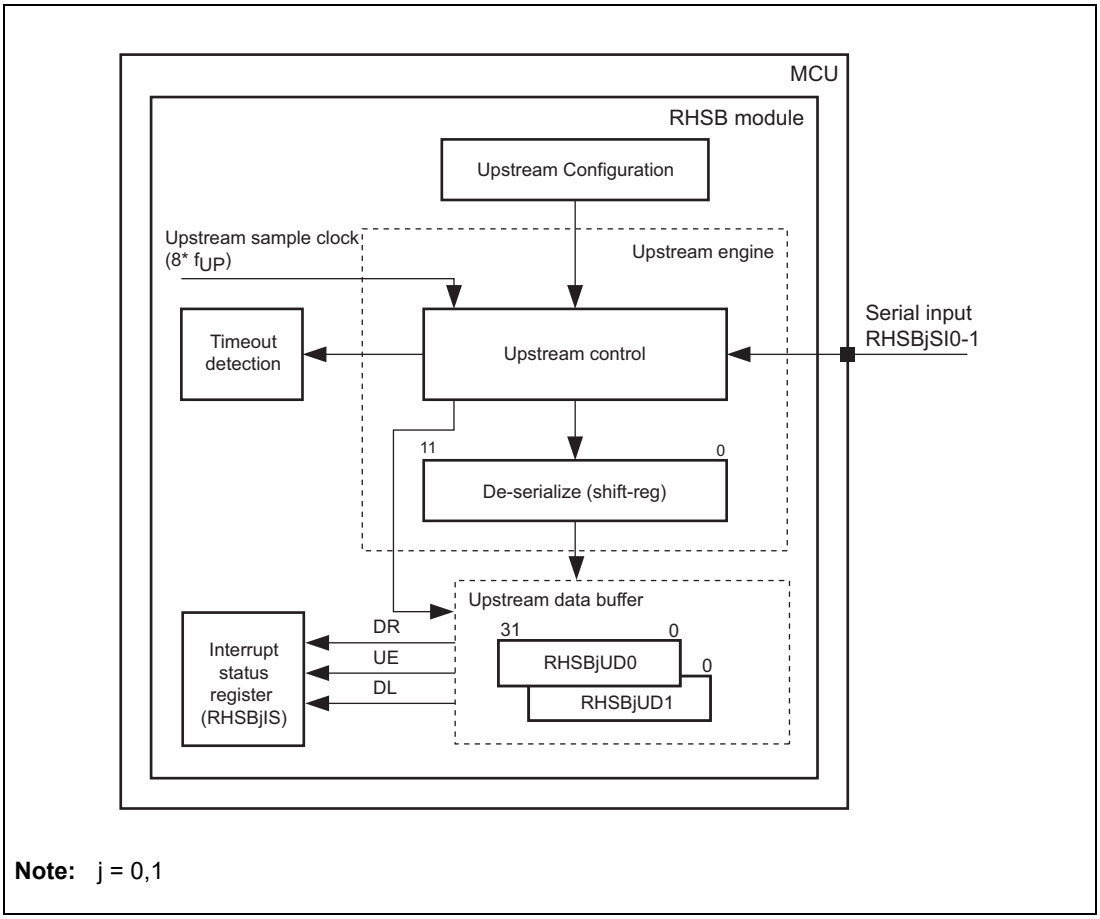


Figure 18.24 Upstream Block Diagram

Upstream Configuration

This block represents the upstream related configuration parameters and the upstream control interface handled by the application software.

Upstream Control

This block provides the main function of the upstream engine. It selects the active serial data input line (RHSBjSI0-1 (j = 0, 1)) and the corresponding configuration, provides the start bit detection, schedules the sampling and observes the frame syntax. In addition this block controls the storage of the received information in the upstream data registers (RHSBjUDi) and provides the required decoding status.

De-serialize

The shift register stores the received payload and provides this information for storage in the upstream data registers.

Upstream Data Buffer

This block includes the upstream data registers and the control logic to select one RHSBjUDi for updating. In addition the upstream related status flags in the RHSBjIS register are controlled by this block.

18.4.2.1 Upstream Modes

The RHSB module has two upstream modes with different methods for how the slaves are connected to the serial input lines. These upstream modes can be configured by RHSBjUCR.RHSBjUMS:

- Dedicated input mode
  - Each slave is connected to an individual serial input line.
  - The active serial input line is selected by RHSBjUCS.RHSBjACC.
- Shared input mode
  - All slaves are connected on the serial input line 0.
  - The active serial input line is always line 0 independently of the value in RHSBjUCS.RHSBjACC.

Note when using the Shared input mode, the unused serial input lines are not required to be configured as RHSB inputs in the MCU port function registers.

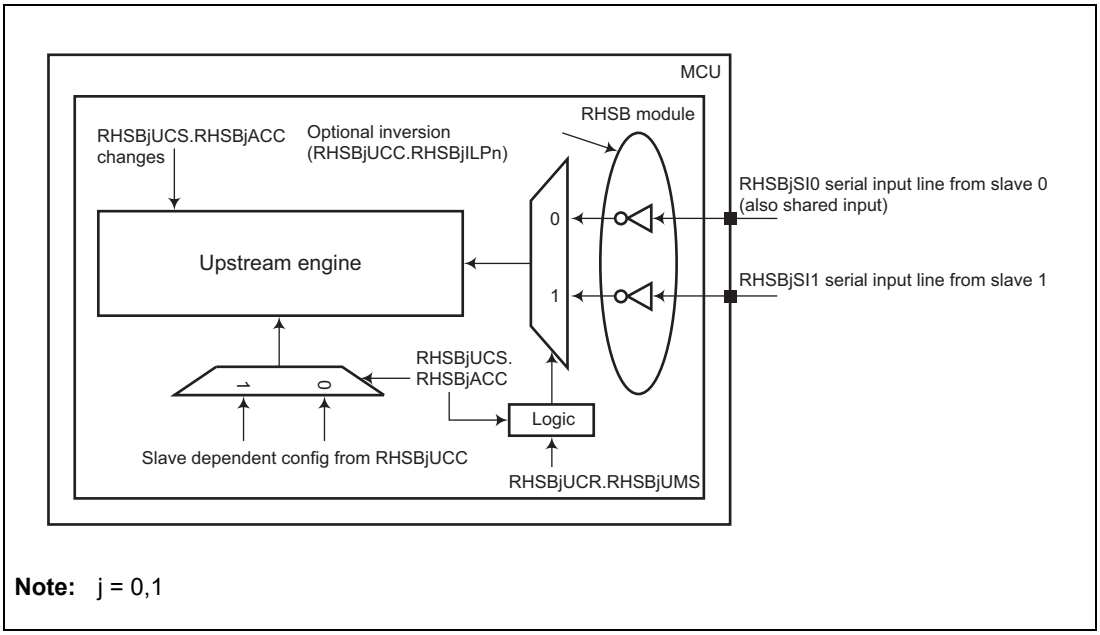


Figure 18.25 Slave Connection and Selection for Upstream Communication

Figure 18.25 illustrates how the serial input lines are connected to the upstream engine. The shown 'Logic' ensures that in shared mode always input line 0 is used, independent from the value of RHSBjUCS.RHSBjACC. The shown 'RHSBjUCS.RHSBjACC changes' information is used by the upstream engine to abort an ongoing reception when the selected channel changes.

18.4.2.2 Individual Slave Configuration

The RHSB module supports for each slave an individual upstream configuration. The usage of different channel configurations is independent from the upstream mode (RHSBjUCR.RHSBjUMS). For each channel the user can program via the RHSBjUCC register:

- The frame type (8 or 12 bit format)
- The number of stop bits (2 or 3)
- The parity type (even or odd)
- The upstream bit rate (please refer to **Section 18.4.2.5, Upstream Bit Rates** for details)

The serial input line polarity is independent from RHSBjUCS.RHSBjACC. The inversion function configured by RHSBjUCC.RHSBjILPn is fixed and assigned to a serial input line n. Thus, RHSBjUCC.RHSBjILP0 is used for all two channels in shared input mode.

The configuration used by the upstream engine to decode upstream frame is defined by RHSBjUCS.RHSBjACC.

There are two ways to change the active upstream configuration:

- Direct request for configuration change by writing to RHSBjUCS.RHSBjACC.
- Request of a command transmission with upstream data request by writing RHSBjDTC.RHSBjCTR to 11_B.

18.4.2.3 Frame Format Types

The RHSB module supports two frame types for upstream communication which only differs in the number of data bits (8 or 12 bit). Other configuration options are not influenced by the frame type.

When the parity type is even, the number of 1_B bits inside the parity covered area is even (0, 2, 4, 6, 8, 10 and 12).

When the parity type is odd, the number of 1_B bits inside the parity covered area is odd (1, 3, 5, 7, 9, 11 and 13).

Figure 18.26 and **Figure 18.27** show an upstream frame with non-inverted data line.

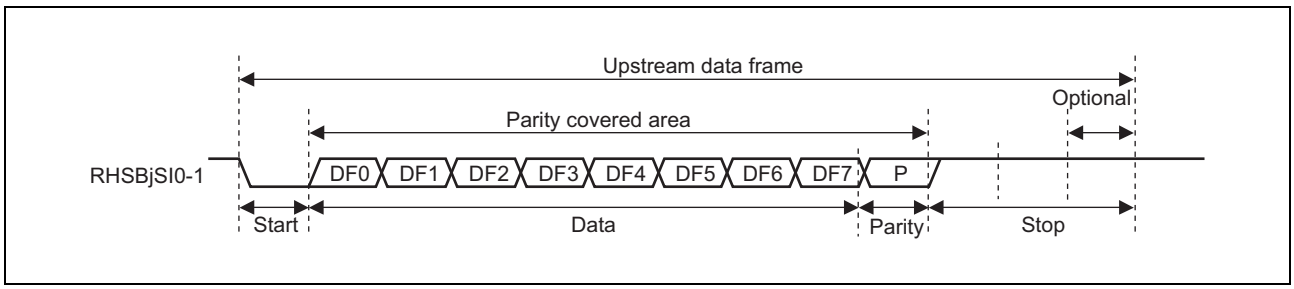


Figure 18.26 Upstream Frame Format with 8 Data Bits (12 or 13 Bit Frame)

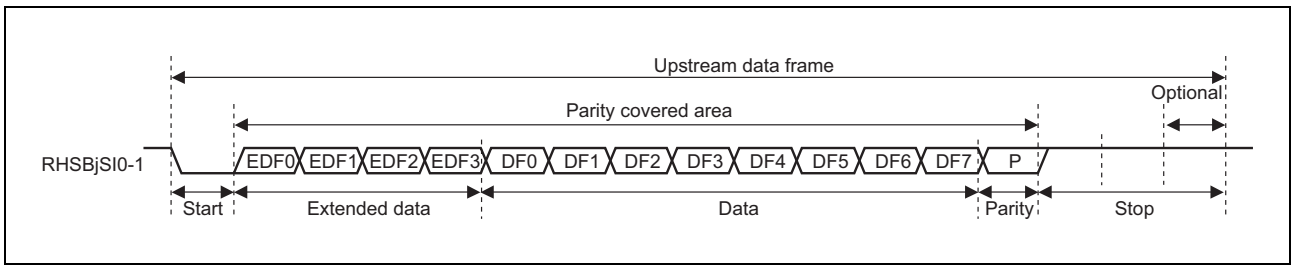


Figure 18.27 Upstream Frame Format with 12 Data Bits (16 or 17 Bit Frame)

18.4.2.4 Frame Storing

The RHSB module offers two upstream data registers (RHSBjUDi) where the upstream engine stores its decoding results. There are two methods selectable in RHSBjUCR.RHSBjFSM defining how the RHSBjUDi is selected for updating.

Note that the decoding of an ongoing reception is aborted when RHSBjUCS.RHSBjACC changes. In this case no decoding result is stored.

(1) Normal storage method

This method is used when RHSBjUCR.RHSBjFSM is 0_B.

The storage strategy is identical for frames of 8-bit and 12-bit format.

The active channel configuration (RHSBjUCS.RHSBjACC) defines the upstream data register to store the decoding results. The target can be identified by this relation:

- when RHSBjUCS.RHSBjACC is 00_B, RHSBjUD0 is updated
- when RHSBjUCS.RHSBjACC is 01_B, RHSBjUD1 is updated

(2) Addressed storage method

This method is used when RHSBjUCR.RHSBjFSM is 1_B.

When the frame type is configured as 8-bit format, the target is always RHSBjUD0.

When the frame type is configures as 12-bit format, the target depends on the received extended data field (EDF, see **Figure 18.27**). The target can be identified by this relation:

- when RHSBjEDF[3:2] is 00_B, RHSBjUD0 is updated
- when RHSBjEDF[3:2] is 01_B, RHSBjUD1 is updated

Note also in case of a faulty reception (parity or stop bit error detected) the received extended data field is used to identify the target for updating the error flags.

18.4.2.5 Upstream Bit Rates

The upstream bit rate (f_{UP}) is derived from the downstream bit rate (f_{DW}).

The bit rate is configured for each channel individually by RHSBjUCC.RHSBjUBRn.

With (x = RHSBjUCC.RHSBjUBRn) the upstream bit rate used by channel n can be calculated as a function of the downstream bit rate such that:

$f_{UP} = f_{DW} / 2^x$

Because the RHSB module samples each upstream bit 8 times, the maximal upstream bit rate is 1/8 of the RHSB engine clock source frequency ($f_{PE}/8$). However, set this RHSB module to 16-frequency division at 5 MHz or lower.

18.4.2.6 Update of Decoding Status

The upstream engine updates one of the RHSBjUDi registers after the decoding of an upstream frame has been completed (after the last stop bit).

When the decoder has detected a parity or stop bit error, the related error flags (RHSBjFERR or RHSBjPERR or both) are set in the corresponding RHSBjUDi register.

When the decoder has not detected parity nor stop bit errors, this frame is judged as a valid frame and these parts of the corresponding RHSBjUDi are updated:

- New data flag (RHSBjND) is set to 1_B
- Data field (RHSBjDF) is updated
- Extended data field (RHSBjEDF) is updated in case of 12-bit frames
- Data lost flag (RHSBjDL) is set to 1_B if the new data flag was already 1_B

18.4.3 Timeout Detection

The concept of timeout detection realized by the RHSB module is linked to the concept of “command transmission with data request”. The concept assumes this flow:

Step 1: Master requests data from slave x by sending a command frame.

Step 2: Slave x answers the master by sending the requested data on the upstream channel.

From an application point of view the master expects to receive an answer from slave x within a certain period (timeout period). This expectation is processed by the timeout detection function.

To use the timeout detection function the user should enable the upstream reception (RHSBjUCR.RHSBjUE is 1_B) and should enable the timeout function (RHSBjUCR.RHSBjTOE is 1_B).

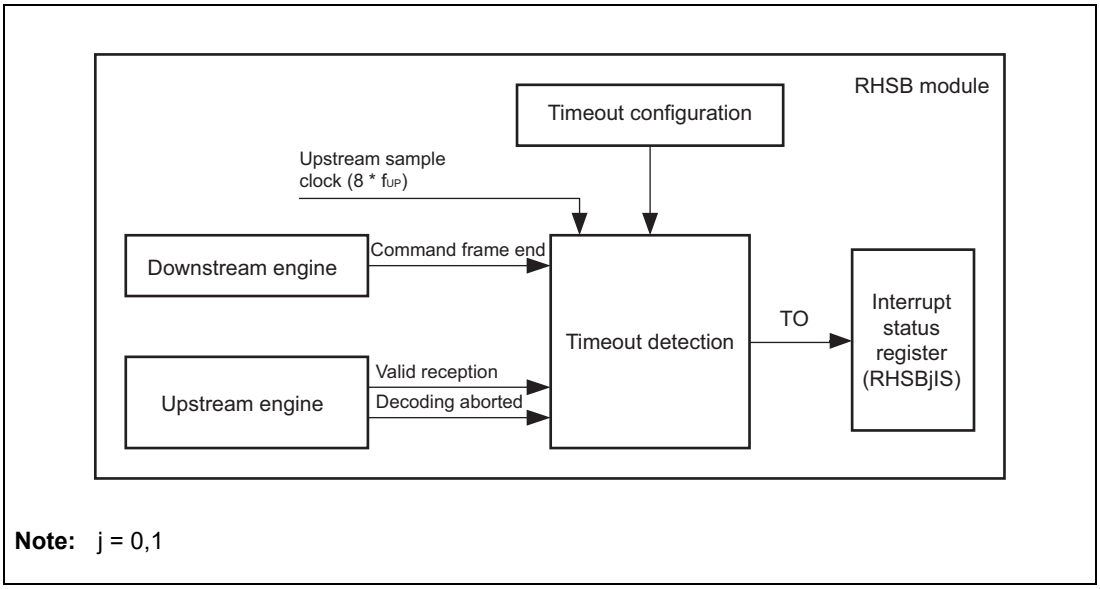


Figure 18.28 Timeout Block Diagram

The diagram in **Figure 18.28** illustrates the scope of timeout detection block. It is controlled by the downstream and upstream engine. The timeout detection controls RHSBjIS.RHSBjTOF. The detection of a timeout has no influence to an ongoing reception.

Note that the timeout detection cannot distinguish between the reception due to the request and other receptions. For example if the shared input mode configuration is used and two slaves are transmitting upstream frames during the timeout period. The user has to ensure that in fault free operation only one upstream reception can happen during the timeout period.

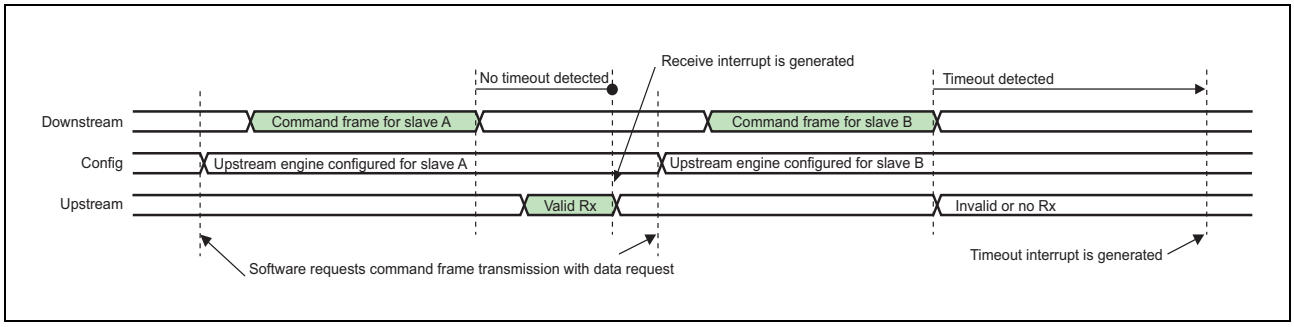


Figure 18.29 Example of Timeout Detection

The example shown in **Figure 18.29** illustrates the principle timing of the timeout detection.

In the first case the user issues a command frame transmission with data request for slave A. The downstream configuration is updated, and then a command frame is scheduled for transmission. After the end of transmission the timeout period starts. The valid response of the slave is decoded before the timeout period ends.

In the second case the user issues a command frame transmission with data request for slave B. In this case the timeout period ends before a valid response was decoded.

18.4.3.1 Command Transmission with Remote Data Request

To perform a command transmission with data request the user has to follow this flow:

- Writing the command data to the RHSBjDCD register
- Define the target slave by writing to RHSBjDTC.RHSBjSSCF
- Request command transmission with data request by writing 11_B to RHSBjDTC.RHSBjCTR

Writing of the command to RHSBjDRC.RHSBjCTR triggers in the RHSB module to carry out these actions:

- Update of the active slave configuration (RHSBjUCS.RHSBjACC) with the target slave number (RHSBjDTC.RHSBjSSCF)
- Abort an ongoing reception if the active slave changes
- Transmit the command frame on the downstream channel
- Start timeout counter when the command transmission has been completed

When there is a valid reception on the upstream channel, the RHSB module judges the remote data request as handled and the timeout counter is stopped.

When there is no valid reception on the upstream channel before the timeout counter elapses, the remote data request is judged as failed and the timeout is flagged by setting RHSBjUDi.RHSBjTO and RHSBjIS.RHSBjTOF to 1_B.

18.4.3.2 Timeout Detection Details

The timeout counter uses the nominal upstream bit time for timeout detection. The timeout counter is not influenced by possible resynchronization effects during start bit decoding.

The timeout counter is initialized to the value (RHSBjUCR.RHSBjRTO + 1)*8 (U bit), when the command frame transmission has been completed (end of frame passive phase).

The timeout counter is decremented with every sample period (U bit/8) used by the decoder according to the active slave configuration (RHSBjUCC.RHSBjUBRn).

The timeout counter stops when:

- there is a valid reception decoded on the upstream channel or
- the timeout counter elapse or
- the decoding is aborted (e.g. by changing the RHSBjUCS.RHSBjACC)

Note that there is no synchronization between the downstream frame passive phase end and the upstream sample points. Thus, the first timeout counter decrement can happen between 1_D bit and 1 U bit/8.

18.4.4 Test Mode Operation

The RHSB module is able to support key-on tests when RHSBjGC.RHSBjOPS is TEST.

Aim of the test mode is to transmit data on the downstream channel and receive the same data as an upstream frame. Because the upstream engine can only decode UART-like frames, such pattern needs to be emulated within the SPI-like downstream frame.

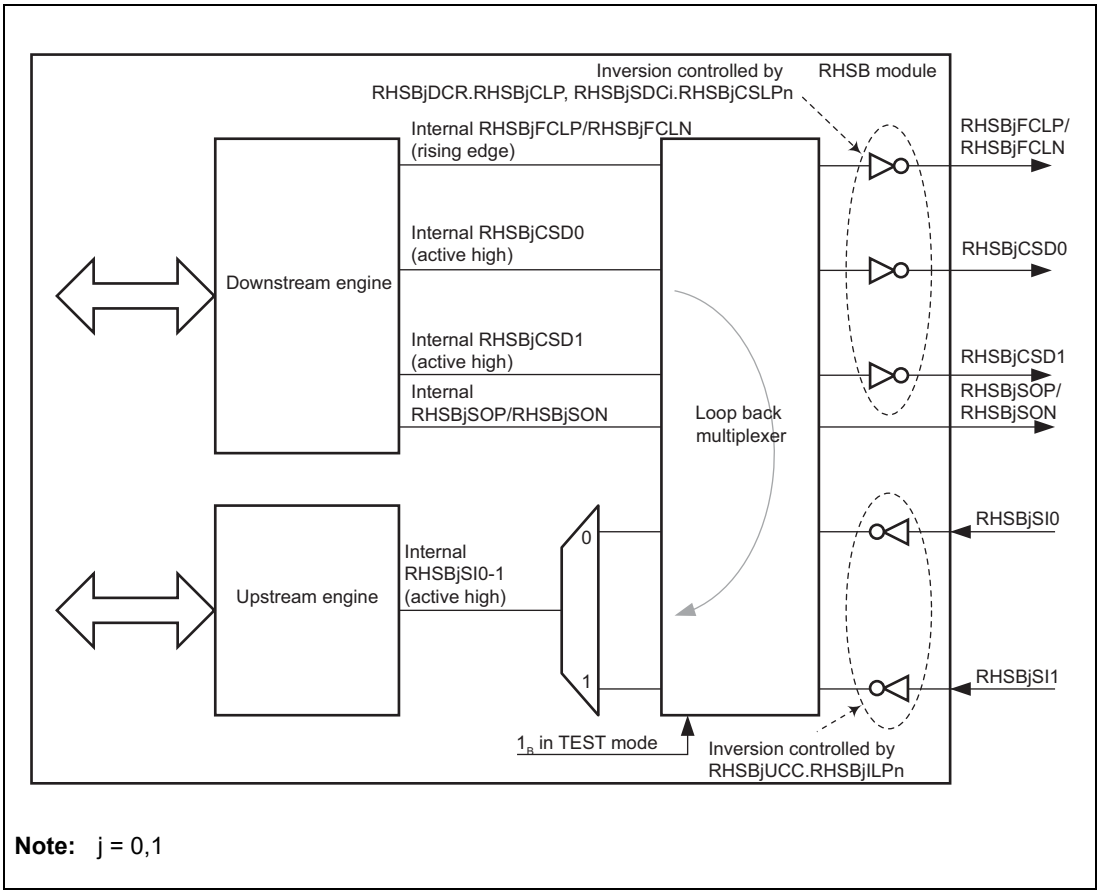
Since the downstream and upstream paths in the RHSB module are independent, the test mode allows good test coverage of all RHSB flags.

While the test mode is active, the serial input lines are disconnected from the upstream engine and the outputs are driven with fixed levels:

- the serial output line (RHSBjSOP/RHSBjSON) is driven with a high level
- the serial clock line (RHSBjFCLP/RHSBjFCLN) is driven with a level depending on the clock line phase (low level when RHSBjDCR.RHSBjCLP is 0_B)
- the chip select lines (RHSBjCSD0-1) are driven with a level depending on the chip select line polarity (low level when RHSBjSDCi.RHSBjCSLPn is 0_B)

Figure 18.30 illustrates the details about the position of the loop-back multiplexer. Only the inversion of the downstream data (RHSBjSDCi.RHSBjSOLPn) is influencing the test mode; all other inversions are out of scope.

The loop-back multiplexer connects the internal serial data (RHSBjSOP/RHSBjSON) signal to one upstream channel. The channel is selected by an active internal chip select. Thus, all two upstream channels can be checked.



Note: j = 0,1

Figure 18.30 Loop-Back Connection in Test Mode

The test mode should be carried out under the following conditions:

- The downstream bit rate divider should be fDW = f_{PE}/8 (this results in the maximum bit rate of 10 MBit/s when using 80 MHz RHSB engine clock source frequency). The user should set RHSBjDCR.RHSBjDBR to 7_D.
- The upstream bit rate should be equal to the downstream bit rate ($f_{UP} = f_{DW}$). The user should set RHSBjUCC.RHSBjUBRn to 0_D for all two upstream channels.

18.4.4.1 Test Mode Data Generation

Figure 18.31 shows an example of how the bits of a command frame are received by the upstream channel. The example uses non inverted downstream data without assertion/deassertion phases and assumes a 12-bit format with two stop bits.

On the downstream data line a command frame is transmitted for slave 1, the receiver has to be connected to the same slave by setting UCS.ACC to 01_B.

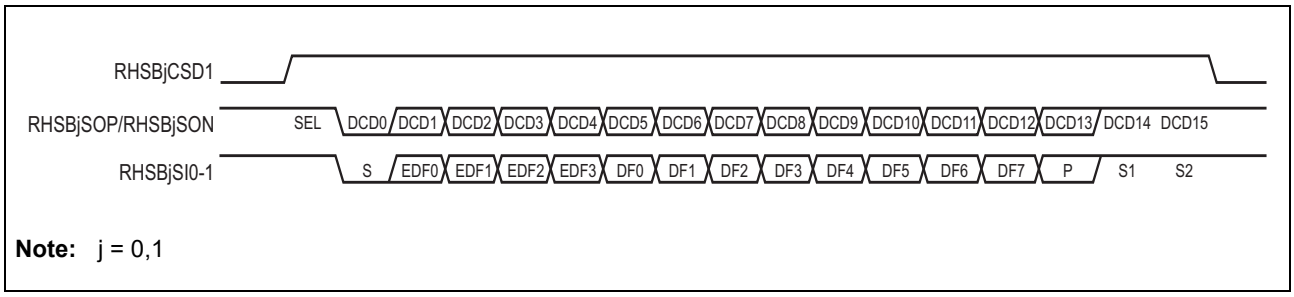


Figure 18.31 Example of How a Command Frame is Received by the Upstream Engine

By writing pattern into the RHSBjDCD register the user is able to check the parity logic and the stop bit error detection. In addition it is possible to check all 32 bits of the RHSBjDCD register, the command length configuration, the assertion/deassertion phase lengths and the data inversion.

Similar checks are also possible with data frames. Note that only the data for the select slave is received, even if multiple slaves are addressed in the data frame.

18.5 Interrupts

The RHSB module has 9 sources which can be used for interrupt request generation.

The interrupt status flags are cleared by the user when writing the flags to 0_B. Writing a 1_B has no effect. For some interrupt flags there are additional functional clear conditions defined to allow an optimized software flow. These interrupt flags are marked by * in the interrupt source lists in **Section 18.5.1** and **Section 18.5.2**.

When the RHSB module is in the RESET state or CONFIG state, an interrupt request is not generated.

18.5.1 Downstream Related Interrupts (Transmission)

Interrupt line 0

Data frame transmission started (RHSBjDTSF) *

Data frame transmission done (RHSBjDTF) *

Interrupt line 1

Command frame transmission done (RHSBjCTF) *

Transmission started (RHSBjTSF)

Interrupt line 2

Emergency frame transmission done (RHSBjETF)

18.5.2 Upstream Related Interrupts (Reception)

Interrupt line 3

Data received (RHSBjDRF) *

Interrupt line 4

Upstream error (RHSBjUEF) *

Timeout detected (RHSBjTOF)

Data lost (RHSBjDLF) *

By reading the RHSBjUSS register the user gets informed about the complete upstream status.

18.5.3 Interrupt Request Details

There is an individual interrupt status flag in the RHSBjIS register (RHSBjIS.RHSBjxF) and an individual interrupt enable bit in the RHSBjIC register (RHSBjIC.RHSBjxIE) for each interrupt source (x).

The function of the interrupt status is independent form the interrupt enable. The interrupt enable controls only if an interrupt request is made to the MCU's interrupt controller.

Figure 18.32 illustrates how an interrupt request is generated and the realization of the interrupt grouping.

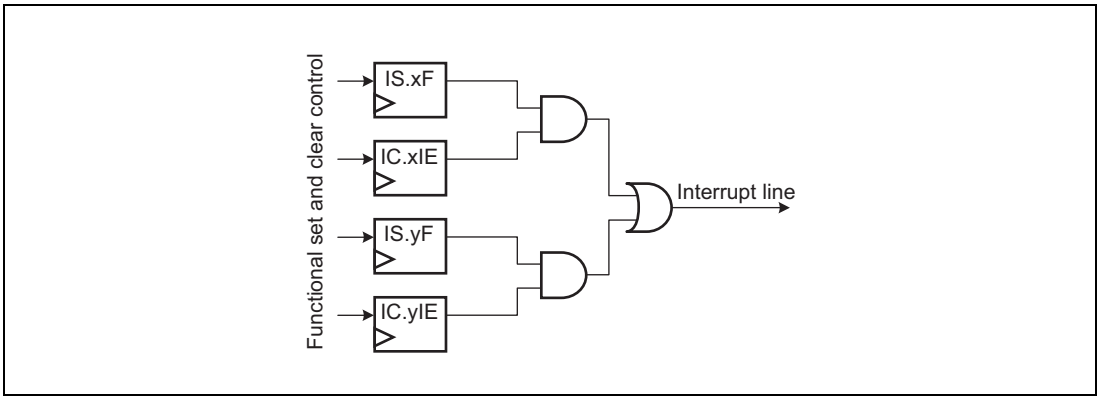


Figure 18.32 Interrupt Grouping (Example of Two Interrupts x and y)

18.6 DMA Capability

18.6.1 DMA Usage for Downstream Data Transmission

The downstream data DMA function is intended to serve applications where pre-calculated data tables are stored in the memory (RAM or FLASH) defining the pattern the RHSB module has to transfer to slaves.

A downstream data DMA transfer is started when RHSBjIS.RHSBjDTSF is 1_B and DMA is enabled by RHSBjGC.RHSBjDDE.

The user should not use RHSBjIS.RHSBjDTSF as interrupt source (RHSBjIC.RHSBjDTSIE should be set to 0_B) when using downstream data DMA. In addition the user should not write to the downstream data registers (RHSBjDDRi) nor set RHSBjIS.RHSBjDTSF to 0_B.

In **Figure 18.33**, case 1 shows normal DMA transfer of downstream data for transmission. In case 2, the DMAC is not able to supply downstream data within the repetition cycle, the DMAC retransmits the last data that it wrote. Accordingly, define a repetition cycle that is longer than the time the DMAC will take to supply the downstream data.

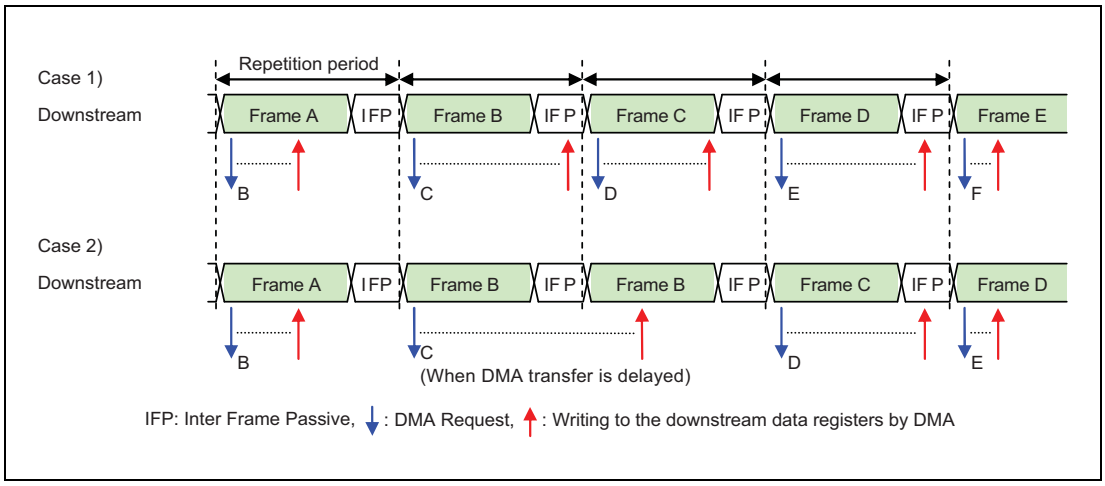


Figure 18.33 Downstream Data DMA Timing

18.6.1.1 Configuring and Enabling Flow to Use Downstream data DMA

These configurations should be done during CONFIG state:

- Set RHSBjDCR.RHSBjDMS to 01_B (triggered mode)
- Set RHSBjIC.RHSBjDTSIE to 0_B
- Set all other configuration parameters according to the application requirements
- Configure the DMA controller to serve the required data to the RHSB module
  - Target in the RHSB module is the RHSBjDDR0 register
  - Number of transferred data words depends on RHSBjDCR.RHSBjSLS

These actions can be done in ACTIVE state according to the application requirements:

- Write initial data to RHSBjDDRi registers

  - The first DMA transfer after setting RHSBjDTC.RHSBjDTE to 1_B is requested after a data frame with this initial data has been transmitted

- Enable or disable the downstream data transmission by setting RHSBjDTC.RHSBjDTE or RHSBjDTC.RHSBjTSR to 1_B.
- Enable or disable the downstream data DMA function (RHSBjGC.RHSBjDDE)

While the downstream data DMA is enabled, RHSBjIS.RHSBjDTSF equal to 1_B indicates a pending DMA transfer.

18.6.1.2 Data Layout in Memory for Downstream Data DMA

The layout of the data to be transferred from the memory to the RHSBjDDRi registers of the RHSB module depends on the data frame configuration in RHSBjDCR.RHSBjSLS.

Because the DMA request (RHSBjIS.RHSBjDTSF) is cleared by writing to the downstream data registers (RHSBjDDRi), the user has to ensure that the transferred amount of data fits to the required data for transmission (refer to **Section 18.4.1.10, Data Update and Data Frame Transmission Request** for details about the request mapping). The RHSB module allows writing more than the required data to the RHSBjDDRi registers. Therefore, there is no functional restriction when the DMA controller supports only 32 bit transfers.

Note when using 16 bit transfers the user has to ensure that the upper part (bits [31:16]) of the downstream data registers are written first.

18.6.2 DMA Usage for Downstream Command Transmission

The downstream command DMA function is intended to serve applications where command data tables are stored in the memory (RAM or FLASH) the RHSB module has to transfer to slaves.

A downstream command DMA transfer is started when RHSBjIS.RHSBjCTF is 1_B and DMA is enabled by RHSBjGC.RHSBjDCDE.

The user should not use RHSBjIS.RHSBjCTF as interrupt source (RHSBjIC.RHSBjCTIE should be set to 0_B) when using downstream command DMA. In addition the user should not write to the downstream command data register (RHSBjDCD) nor downstream transmission control register (RHSBjDTC) nor set RHSBjIS.RHSBjCTF to 0_B.

18.6.2.1 Configuring and Enabling Flow to Use Downstream Command DMA

These configurations should be done during CONFIG state:

- Set RHSBjIC.RHSBjCTIE to 0_B
- Set all other configuration parameters according to the application requirements
- Configure the DMA controller to serve the required data to the RHSB module
  - Targets in the RHSB module are the RHSBjDCD register and RHSBjDTC register
  - Number of transferred command bits depends on RHSBjDTC.RHSBjNCB

These actions can be done in ACTIVE state according to the application requirements:

- Write command data to RHSBjDCD register and configuration parameters to RHSBjDTC register
- Enable or disable the downstream command DMA function (RHSBjGC.RHSBjDCDE)

While the downstream command DMA is enabled, RHSBjIS.RHSBjCTF equal to 1_B indicates a pending DMA transfer.

18.6.2.2 Command Data Layout in Memory for Downstream Command DMA

The layout of the command data to be transferred from the memory to the RHSBjDCD register of the RHSB module depends on the command frame configuration in RHSBjDTC.RHSBjNCB.

Because the DMA request (RHSBjIS.RHSBjCTF) is cleared by writing to the command transmission request (RHSBjDTC.RHSBjCTR), the user has to ensure that the transferred amount of command data fits to the required command for transmission. The RHSB module allows writing more than the required data to the RHSBjDCD register. Therefore, there is no functional restriction when the DMA controller supports only 32 bit transfers.

Note when using 16 bit transfers the user has to ensure that the upper part (bits [31:16]) of the downstream command data register is written first.

18.6.3 DMA Usage for Upstream Data Reception

The upstream DMA function is intended to store a set of upstream frames into the memory.

An upstream DMA transfer is started when RHSBjIS.RHSBjDRF is 1_B and DMA is enabled by RHSBjGC.RHSBjUDE.

The user should not use RHSBjIS.RHSBjDRF as interrupt source (RHSBjIC.RHSBjDRIE should be set to 0_B) when using upstream DMA. In addition the user should not read from the upstream data read registers (RHSBjUDR) nor set an RHSBjUDi.RHSBjND to 0_B nor set RHSBjIS.RHSBjDRF to 0_B.

When the DMA controller is not able to process the new received data timely, a received message might be lost because it is no longer available in the RHSBjUDR (even if still available in the related RHSBjUDi). The software gets informed about the overrun by the RHSBjUDR.RHSBjNDS bits and about a data lost by RHSBjUDi.RHSBjDL.

18.6.3.1 Configuring and Enabling Flow to Use Upstream DMA

These configurations should be done during CONFIG state:

- Set RHSBjUCR.RHSBjUE to 1_B
- Set RHSBjIC.RHSBjDRIE to 0_B
- Set all other configuration parameters according to the application requirements
- Configure the DMA controller to serve the required data to the RHSB module
  - Source in the RHSB module is RHSBjUDR
  - Number of transferred data (8, 16, 32 bits) depends on the application requirements

These actions can be done in ACTIVE state according to the application requirements:

- Request data from the slaves (e.g. by command frame)
- Enable or disable the upstream DMA function (RHSBjGC.RHSBjUDE)
- Evaluate the upstream error interrupts to handle upstream problems

While the upstream DMA is enabled, RHSBjIS.RHSBjDRF equal to 1_B indicates a pending DMA transfer.

18.6.3.2 Data Layout in Memory for Upstream DMA

The DMA transfer should only use the RHSBjUDR as source for the DMA transfer. It depends on the application requirements whether all 32 bits of the RHSBjUDR are transferred to the memory or only a part (8 or 16 bits).

When the DMA controller reads from the RHSBjUDR, the related new data flag in an RHSBjUDi register is set to 0_B. In addition RHSBjIS.RHSBjDRF is set to 0_B. Note that the error flags in the RHSBjUDi registers are not touched by the DMA transfer.

18.7 Cross Bar (XBAR)

18.7.1 Overview

XBAR is configured of the following functions.

- RHSB XBAR

The RHSB XBAR selects signals from the ATU-IV and APA with the multiplexer and outputs arbitrary signals to the RHSB. The RHSB has two channels , each of which has a XBAR. Each XBAR has four 16-bit sub XBARs. To output signals to the RHSB is specified by setting the MSBiCRjH and MSBiCRjL bits. RHSB0 and RHSB1 select the different signals in XBAR, where i = 0 for RHSB0 and i = 1 for RHSB1.

18.7.2 Module Configuration

18.7.2.1 RHSB XBAR Configuration

The RHSB cross bar is selection logic that connects between ATU- IV and RHSB. The RHSB cross bar has four 16-bit sub XBARs for each RHSB channel.

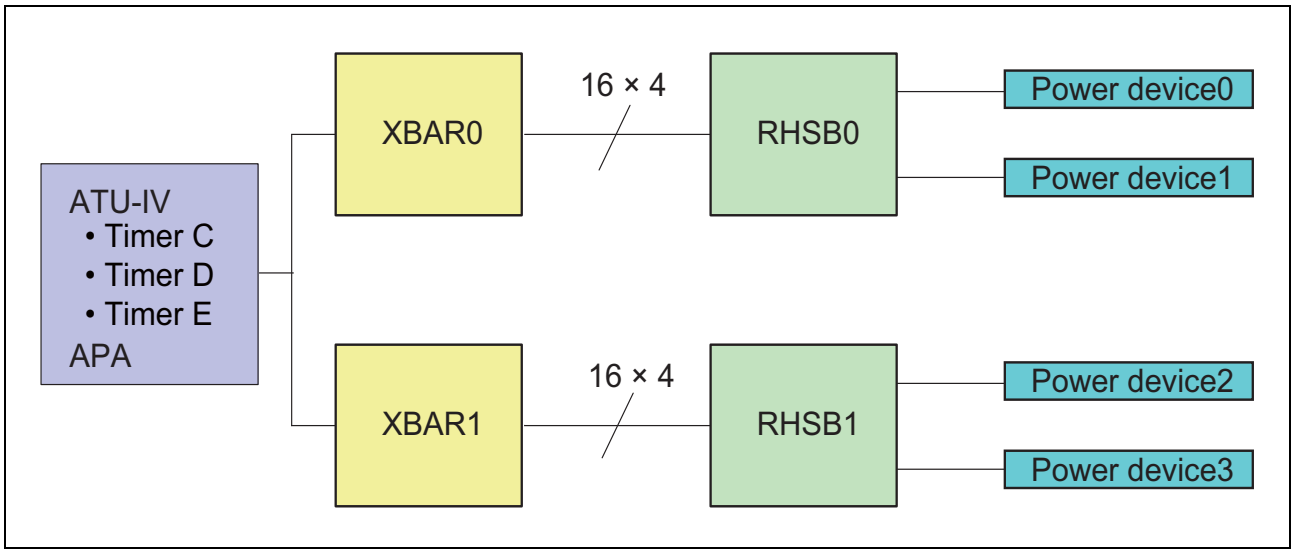


Figure 18.34 RHSB XBAR Configuration

18.7.3 Register Specifications

18.7.3.1 List of Registers

The following tables show address allocation of RHSB XBAR registers.

Table 18.29 RHSB0 XBAR Registers

Address	Allocation of Registers
FFED D000$_H$	RHSBG0CR0H
FFED D004$_H$	RHSBG0CR0L
FFED D008$_H$	RHSBG0CR1H
FFED D00C$_H$	RHSBG0CR1L
FFED D010$_H$	RHSBG0CR2H
FFED D014$_H$	RHSBG0CR2L
FFED D018$_H$	RHSBG0CR3H
FFED D01C$_H$	RHSBG0CR3L

Table 18.30 RHSB1 XBAR Registers

Address	Allocation of Registers
FFED D800$_H$	RHSBG1CR0H
FFED D804$_H$	RHSBG1CR0L
FFED D808$_H$	RHSBG1CR1H
FFED D80C$_H$	RHSBG1CR1L
FFED D810$_H$	RHSBG1CR2H
FFED D814$_H$	RHSBG1CR2L
FFED D818$_H$	RHSBG1CR3H
FFED D81C$_H$	RHSBG1CR3L

18.7.3.2 RHSBGiCRjH — Microsecond Bus Control Register H

RHSBGiCRjH is a 32-bit readable/writable register that selects upper bits of signals to be output to the RHSB from signals received from the ATU and APA. This register supports ATU timer C, timer D, timer E, and APA signals.

Do not change the setting of bits in this register while the ATU timer C, timer D, timer E, or APA is working.

RHSBGiCRjH can be read and written in byte units.

RHSBGiCRjH is initialized to 0000 0000_H by a reset.

i = 0, 1 (number of RHSB channels)

j = 0 to 3 (number of sub XBARs)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MiMDj15				MiMDj14				MiMDj13				MiMDj12			
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MiMDj11				MiMDj10				MiMDj9				MiMDj8			
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 18.31 RHSBGiCRjH Register Contents (1/4)

Bit Position	Bit Name	Function
31 to 28	MiMDj15	These bits select a signal of the fifteenth bit. 0000: TIOC0[3] is selected. 0001: TIOC2[3] is selected. (i = 0)/TIOC5[3] is selected. (i = 1) 0010: TIOC3[3] is selected. (i = 0)/TIOC6[3] is selected. (i = 1) 0011: TIOC4[3] is selected. (i = 0)/TIOC7[3] is selected. (i = 1) 0100: TOD0B[3] is selected. 0101: TOD1B[3] is selected. 0110: TOD2B[3] is selected. (i = 0)/TOD4B[3] is selected. (i = 1) 0111: TOD3B[3] is selected. (i = 0)/TOD5B[3] is selected. (i = 1) 1000: TOE0[3] is selected. 1001: TOE1[3] is selected. (i = 0)/TOE4[3] is selected. (i = 1) 1010: TOE2[3] is selected. (i = 0)/TOE5[3] is selected. (i = 1) 1011: TOE3[3] is selected. (i = 0)/TOE6[3] is selected. (i = 1) 1100: TOD3A[3] is selected. (i = 0)/TOD5A[3] is selected. (i = 1) 1101: TOD0A[3] is selected. 1110: APA[7] is selected. 1111: APA[15] is selected.

Table 18.31 RHSBGiCRjH Register Contents (2/4)

Bit Position	Bit Name	Function
27 to 24	MiMDj14	These bits select a signal of the fourteenth bit. 0000: TIOC0[2] is selected. 0001: TIOC2[2] is selected. (i = 0)/TIOC5[2] is selected. (i = 1) 0010: TIOC3[2] is selected. (i = 0)/TIOC6[2] is selected. (i = 1) 0011: TIOC4[2] is selected. (i = 0)/TIOC7[2] is selected. (i = 1) 0100: TOD0B[2] is selected. 0101: TOD1B[2] is selected. 0110: TOD2B[2] is selected. (i = 0)/TOD4B[2] is selected. (i = 1) 0111: TOD3B[2] is selected. (i = 0)/TOD5B[2] is selected. (i = 1) 1000: TOE0[2] is selected. 1001: TOE1[2] is selected. (i = 0)/TOE4[2] is selected. (i = 1) 1010: TOE2[2] is selected. (i = 0)/TOE5[2] is selected. (i = 1) 1011: TOE3[2] is selected. (i = 0)/TOE6[2] is selected. (i = 1) 1100: TOD3A[2] is selected. (i = 0)/TOD5A[2] is selected. (i = 1) 1101: TOD0A[2] is selected. 1110: APA[6] is selected. 1111: APA[14] is selected.
23 to 20	MiMDj13	These bits select a signal of the thirteenth bit. 0000: TIOC0[1] is selected. 0001: TIOC2[1] is selected. (i = 0)/TIOC5[1] is selected. (i = 1) 0010: TIOC3[1] is selected. (i = 0)/TIOC6[1] is selected. (i = 1) 0011: TIOC4[1] is selected. (i = 0)/TIOC7[1] is selected. (i = 1) 0100: TOD0B[1] is selected. 0101: TOD1B[1] is selected. 0110: TOD2B[1] is selected. (i = 0)/TOD4B[1] is selected. (i = 1) 0111: TOD3B[1] is selected. (i = 0)/TOD5B[1] is selected. (i = 1) 1000: TOE0[1] is selected. 1001: TOE1[1] is selected. (i = 0)/TOE4[1] is selected. (i = 1) 1010: TOE2[1] is selected. (i = 0)/TOE5[1] is selected. (i = 1) 1011: TOE3[1] is selected. (i = 0)/TOE6[1] is selected. (i = 1) 1100: TOD3A[1] is selected. (i = 0)/TOD5A[1] is selected. (i = 1) 1101: TOD0A[1] is selected. 1110: APA[5] is selected. 1111: APA[13] is selected.
19 to 16	MiMDj12	These bits select a signal of the twelfth bit. 0000: TIOC0[0] is selected. 0001: TIOC2[0] is selected. (i = 0)/TIOC5[0] is selected. (i = 1) 0010: TIOC3[0] is selected. (i = 0)/TIOC6[0] is selected. (i = 1) 0011: TIOC4[0] is selected. (i = 0)/TIOC7[0] is selected. (i = 1) 0100: TOD0B[0] is selected. 0101: TOD1B[0] is selected. 0110: TOD2B[0] is selected. (i = 0)/TOD4B[0] is selected. (i = 1) 0111: TOD3B[0] is selected. (i = 0)/TOD5B[0] is selected. (i = 1) 1000: TOE0[0] is selected. 1001: TOE1[0] is selected. (i = 0)/TOE4[0] is selected. (i = 1) 1010: TOE2[0] is selected. (i = 0)/TOE5[0] is selected. (i = 1) 1011: TOE3[0] is selected. (i = 0)/TOE6[0] is selected. (i = 1) 1100: TOD3A[0] is selected. (i = 0)/TOD5A[0] is selected. (i = 1) 1101: TOD0A[0] is selected. 1110: APA[4] is selected. 1111: APA[12] is selected.

Table 18.31 RHSBGiCRjH Register Contents (3/4)

Bit Position	Bit Name	Function
15 to 12	MiMDj11	These bits select a signal of the eleventh bit. 0000: TIOC0[3] is selected. 0001: TIOC1[3] is selected. 0010: TIOC3[3] is selected. (i = 0)/TIOC6[3] is selected. (i = 1) 0011: TIOC4[3] is selected. (i = 0)/TIOC7[3] is selected. (i = 1) 0100: TOD0B[3] is selected. 0101: TOD1B[3] is selected. 0110: TOD2B[3] is selected. (i = 0)/TOD4B[3] is selected. (i = 1) 0111: TOD3B[3] is selected. (i = 0)/TOD5B[3] is selected. (i = 1) 1000: TOE0[3] is selected. 1001: TOE1[3] is selected. (i = 0)/TOE4[3] is selected. (i = 1) 1010: TOE2[3] is selected. (i = 0)/TOE5[3] is selected. (i = 1) 1011: TOE3[3] is selected. (i = 0)/TOE6[3] is selected. (i = 1) 1100: TOD2A[3] is selected. (i = 0)/TOD4A[3] is selected. (i = 1) 1101: TOD3A[3] is selected. (i = 0)/TOD5A[3] is selected. (i = 1) 1110: APA[3] is selected. 1111: APA[11] is selected.
11 to 8	MiMDj10	These bits select a signal of the tenth bit. 0000: TIOC0[2] is selected. 0001: TIOC1[2] is selected. 0010: TIOC3[2] is selected. (i = 0)/TIOC6[2] is selected. (i = 1) 0011: TIOC4[2] is selected. (i = 0)/TIOC7[2] is selected. (i = 1) 0100: TOD0B[2] is selected. 0101: TOD1B[2] is selected. 0110: TOD2B[2] is selected. (i = 0)/TOD4B[2] is selected. (i = 1) 0111: TOD3B[2] is selected. (i = 0)/TOD5B[2] is selected. (i = 1) 1000: TOE0[2] is selected. 1001: TOE1[2] is selected. (i = 0)/TOE4[2] is selected. (i = 1) 1010: TOE2[2] is selected. (i = 0)/TOE5[2] is selected. (i = 1) 1011: TOE3[2] is selected. (i = 0)/TOE6[2] is selected. (i = 1) 1100: TOD2A[2] is selected. (i = 0)/TOD4A[2] is selected. (i = 1) 1101: TOD3A[2] is selected. (i = 0)/TOD5A[2] is selected. (i = 1) 1110: APA[2] is selected. 1111: APA[10] is selected.
7 to 4	MiMDj9	These bits select a signal of the ninth bit. 0000: TIOC0[1] is selected. 0001: TIOC1[1] is selected. 0010: TIOC3[1] is selected. (i = 0)/TIOC6[1] is selected. (i = 1) 0011: TIOC4[1] is selected. (i = 0)/TIOC7[1] is selected. (i = 1) 0100: TOD0B[1] is selected. 0101: TOD1B[1] is selected. 0110: TOD2B[1] is selected. (i = 0)/TOD4B[1] is selected. (i = 1) 0111: TOD3B[1] is selected. (i = 0)/TOD5B[1] is selected. (i = 1) 1000: TOE0[1] is selected. 1001: TOE1[1] is selected. (i = 0)/TOE4[1] is selected. (i = 1) 1010: TOE2[1] is selected. (i = 0)/TOE5[1] is selected. (i = 1) 1011: TOE3[1] is selected. (i = 0)/TOE6[1] is selected. (i = 1) 1100: TOD2A[1] is selected. (i = 0)/TOD4A[1] is selected. (i = 1) 1101: TOD3A[1] is selected. (i = 0)/TOD5A[1] is selected. (i = 1) 1110: APA[1] is selected. 1111: APA[9] is selected.

Table 18.31 RHSBGiCRjH Register Contents (4/4)

Bit Position	Bit Name	Function
3 to 0	MiMDj8	These bits select a signal of the eighth bit. 0000: TIOC0[0] is selected. 0001: TIOC1[0] is selected. 0010: TIOC3[0] is selected. (i = 0)/TIOC6[0] is selected. (i = 1) 0011: TIOC4[0] is selected. (i = 0)/TIOC7[0] is selected. (i = 1) 0100: TOD0B[0] is selected. 0101: TOD1B[0] is selected. 0110: TOD2B[0] is selected. (i = 0)/TOD4B[0] is selected. (i = 1) 0111: TOD3B[0] is selected. (i = 0)/TOD5B[0] is selected. (i = 1) 1000: TOE0[0] is selected. 1001: TOE1[0] is selected. (i = 0)/TOE4[0] is selected. (i = 1) 1010: TOE2[0] is selected. (i = 0)/TOE5[0] is selected. (i = 1) 1011: TOE3[0] is selected. (i = 0)/TOE6[0] is selected. (i = 1) 1100: TOD2A[0] is selected. (i = 0)/TOD4A[0] is selected. (i = 1) 1101: TOD3A[0] is selected. (i = 0)/TOD5A[0] is selected. (i = 1) 1110: APA[0] is selected. 1111: APA[8] is selected.

18.7.3.3 RHSBGiCRjL — Microsecond Bus Control Register L

RHSBGiCRjL is a 32-bit readable/writable register that selects lower bits of signals to be output to the RHSB from signals received from the ATU and APA. This register supports ATU timer C, timer D, timer E, and APA signals.

Do not change the setting of bits in this register while the ATU timer C, timer D, timer E, or APA is working.

RHSBGiCRjL can be read and written in byte units.

RHSBGiCRjL is initialized to 0000 0000_H by a reset.

i = 0, 1 (number of RHSB channels)

j = 0 to 3 (number of sub XBARs)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MiMDj7				MiMDj6				MiMDj5				MiMDj4			
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MiMDj3				MiMDj2				MiMDj1				MiMDj0			
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 18.32 RHSBGiCRjL Register Contents (1/4)

Bit Position	Bit Name	Function
31 to 28	MiMDj7	These bits select a signal of the seventh bit. 0000: TIOC0[3] is selected. 0001: TIOC1[3] is selected. 0010: TIOC2[3] is selected. (i = 0)/TIOC5[3] is selected. (i = 1) 0011: TIOC4[3] is selected. (i = 0)/TIOC7[3] is selected. (i = 1) 0100: TOD0B[3] is selected. 0101: TOD1B[3] is selected. 0110: TOD2B[3] is selected. (i = 0)/TOD4B[3] is selected. (i = 1) 0111: TOD3B[3] is selected. (i = 0)/TOD5B[3] is selected. (i = 1) 1000: TOE0[3] is selected. 1001: TOE1[3] is selected. (i = 0)/TOE4[3] is selected. (i = 1) 1010: TOE2[3] is selected. (i = 0)/TOE5[3] is selected. (i = 1) 1011: TOE3[3] is selected. (i = 0)/TOE6[3] is selected. (i = 1) 1100: TOD1A[3] is selected. 1101: TOD2A[3] is selected. (i = 0)/TOD4A[3] is selected. (i = 1) 1110: APA[7] is selected. 1111: APA[15] is selected.

Table 18.32 RHSBGiCRjL Register Contents (2/4)

Bit Position	Bit Name	Function
27 to 24	MiMDj6	These bits select a signal of the sixth bit. 0000: TIOC0[2] is selected. 0001: TIOC1[2] is selected. 0010: TIOC2[2] is selected. (i = 0)/TIOC5[2] is selected. (i = 1) 0011: TIOC4[2] is selected. (i = 0)/TIOC7[2] is selected. (i = 1) 0100: TOD0B[2] is selected. 0101: TOD1B[2] is selected. 0110: TOD2B[2] is selected. (i = 0)/TOD4B[2] is selected. (i = 1) 0111: TOD3B[2] is selected. (i = 0)/TOD5B[2] is selected. (i = 1) 1000: TOE0[2] is selected. 1001: TOE1[2] is selected. (i = 0)/TOE4[2] is selected. (i = 1) 1010: TOE2[2] is selected. (i = 0)/TOE5[2] is selected. (i = 1) 1011: TOE3[2] is selected. (i = 0)/TOE6[2] is selected. (i = 1) 1100: TOD1A[2] is selected. 1101: TOD2A[2] is selected. (i = 0)/TOD4A[2] is selected. (i = 1) 1110: APA[6] is selected. 1111: APA[14] is selected.
23 to 20	MiMDj5	These bits select a signal of the fifth bit. 0000: TIOC0[1] is selected. 0001: TIOC1[1] is selected. 0010: TIOC2[1] is selected. (i = 0)/TIOC5[1] is selected. (i = 1) 0011: TIOC4[1] is selected. (i = 0)/TIOC7[1] is selected. (i = 1) 0100: TOD0B[1] is selected. 0101: TOD1B[1] is selected. 0110: TOD2B[1] is selected. (i = 0)/TOD4B[1] is selected. (i = 1) 0111: TOD3B[1] is selected. (i = 0)/TOD5B[1] is selected. (i = 1) 1000: TOE0[1] is selected. 1001: TOE1[1] is selected. (i = 0)/TOE4[1] is selected. (i = 1) 1010: TOE2[1] is selected. (i = 0)/TOE5[1] is selected. (i = 1) 1011: TOE3[1] is selected. (i = 0)/TOE6[1] is selected. (i = 1) 1100: TOD1A[1] is selected. 1101: TOD2A[1] is selected. (i = 0)/TOD4A[1] is selected. (i = 1) 1110: APA[5] is selected. 1111: APA[13] is selected.
19 to 16	MiMDj4	These bits select a signal of the fourth bit. 0000: TIOC0[0] is selected. 0001: TIOC1[0] is selected. 0010: TIOC2[0] is selected. (i = 0)/TIOC5[0] is selected. (i = 1) 0011: TIOC4[0] is selected. (i = 0)/TIOC7[0] is selected. (i = 1) 0100: TOD0B[0] is selected. 0101: TOD1B[0] is selected. 0110: TOD2B[0] is selected. (i = 0)/TOD4B[0] is selected. (i = 1) 0111: TOD3B[0] is selected. (i = 0)/TOD5B[0] is selected. (i = 1) 1000: TOE0[0] is selected. 1001: TOE1[0] is selected. (i = 0)/TOE4[0] is selected. (i = 1) 1010: TOE2[0] is selected. (i = 0)/TOE5[0] is selected. (i = 1) 1011: TOE3[0] is selected. (i = 0)/TOE6[0] is selected. (i = 1) 1100: TOD1A[0] is selected. 1101: TOD2A[0] is selected. (i = 0)/TOD4A[0] is selected. (i = 1) 1110: APA[4] is selected. 1111: APA[12] is selected.

Table 18.32 RHSBGiCRjL Register Contents (3/4)

Bit Position	Bit Name	Function
15 to 12	MiMDj3	These bits select a signal of the third bit. 0000: TIOC0[3] is selected. 0001: TIOC1[3] is selected. 0010: TIOC2[3] is selected. (i = 0)/TIOC5[3] is selected. (i = 1) 0011: TIOC3[3] is selected. (i = 0)/TIOC6[3] is selected. (i = 1) 0100: TOD0B[3] is selected. 0101: TOD1B[3] is selected. 0110: TOD2B[3] is selected. (i = 0)/TOD4B[3] is selected. (i = 1) 0111: TOD3B[3] is selected. (i = 0)/TOD5B[3] is selected. (i = 1) 1000: TOE0[3] is selected. 1001: TOE1[3] is selected. (i = 0)/TOE4[3] is selected. (i = 1) 1010: TOE2[3] is selected. (i = 0)/TOE5[3] is selected. (i = 1) 1011: TOE3[3] is selected. (i = 0)/TOE6[3] is selected. (i = 1) 1100: TOD0A[3] is selected. 1101: TOD1A[3] is selected. 1110: APA[3] is selected. 1111: APA[11] is selected.
11 to 8	MiMDj2	These bits select a signal of the second bit. 0000: TIOC0[2] is selected. 0001: TIOC1[2] is selected. 0010: TIOC2[2] is selected. (i = 0)/TIOC5[2] is selected. (i = 1) 0011: TIOC3[2] is selected. (i = 0)/TIOC6[2] is selected. (i = 1) 0100: TOD0B[2] is selected. 0101: TOD1B[2] is selected. 0110: TOD2B[2] is selected. (i = 0)/TOD4B[2] is selected. (i = 1) 0111: TOD3B[2] is selected. (i = 0)/TOD5B[2] is selected. (i = 1) 1000: TOE0[2] is selected. 1001: TOE1[2] is selected. (i = 0)/TOE4[2] is selected. (i = 1) 1010: TOE2[2] is selected. (i = 0)/TOE5[2] is selected. (i = 1) 1011: TOE3[2] is selected. (i = 0)/TOE6[2] is selected. (i = 1) 1100: TOD0A[2] is selected. 1101: TOD1A[2] is selected. 1110: APA[2] is selected. 1111: APA[10] is selected.
7 to 4	MiMDj1	These bits select a signal of the first bit. 0000: TIOC0[1] is selected. 0001: TIOC1[1] is selected. 0010: TIOC2[1] is selected. (i = 0)/TIOC5[1] is selected. (i = 1) 0011: TIOC3[1] is selected. (i = 0)/TIOC6[1] is selected. (i = 1) 0100: TOD0B[1] is selected. 0101: TOD1B[1] is selected. 0110: TOD2B[1] is selected. (i = 0)/TOD4B[1] is selected. (i = 1) 0111: TOD3B[1] is selected. (i = 0)/TOD5B[1] is selected. (i = 1) 1000: TOE0[1] is selected. 1001: TOE1[1] is selected. (i = 0)/TOE4[1] is selected. (i = 1) 1010: TOE2[1] is selected. (i = 0)/TOE5[1] is selected. (i = 1) 1011: TOE3[1] is selected. (i = 0)/TOE6[1] is selected. (i = 1) 1100: TOD0A[1] is selected. 1101: TOD1A[1] is selected. 1110: APA[1] is selected. 1111: APA[9] is selected.

Table 18.32 RHSBGiCRjL Register Contents (4/4)

Bit Position	Bit Name	Function
3 to 0	MiMDj0	These bits select a signal of the zeroth bit. 0000: TIOC0[0] is selected. 0001: TIOC1[0] is selected. 0010: TIOC2[0] is selected. (i = 0)/TIOC5[0] is selected. (i = 1) 0011: TIOC3[0] is selected. (i = 0)/TIOC6[0] is selected. (i = 1) 0100: TOD0B[0] is selected. 0101: TOD1B[0] is selected. 0110: TOD2B[0] is selected. (i = 0)/TOD4B[0] is selected. (i = 1) 0111: TOD3B[0] is selected. (i = 0)/TOD5B[0] is selected. (i = 1) 1000: TOE0[0] is selected. 1001: TOE1[0] is selected. (i = 0)/TOE4[0] is selected. (i = 1) 1010: TOE2[0] is selected. (i = 0)/TOE5[0] is selected. (i = 1) 1011: TOE3[0] is selected. (i = 0)/TOE6[0] is selected. (i = 1) 1100: TOD0A[0] is selected. 1101: TOD1A[0] is selected. 1110: APA[0] is selected. 1111: APA[8] is selected.

18.7.4 Summary of Operation

Signals received from the ATU-IV and APA are selected by the multiplexer and arbitrary signals are output to the RHSB. The RHSB has two channels , each of which has a XBAR. Each XBAR has four 16-bit sub XBAR. Output to the RHSB can be selected as shown in **Table 18.33** by changing each bit in RHSBGiCRjH and RHSBGiCRjL. Different signals are selected by the cross bar in RHSB0 and RHSB1. RHSB0 supports i = 0 and RHSB1 supports i = 1.

Table 18.33 Selectable Bits of the RHSB XBAR
( ): Specifications of XBAR Connected to RHSB1

Output	Selectable Bits of XBAR Connected to RHSB0															
	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
0	TIOC 00	TIOC 10	TIOC 20 (50)	TIOC 30 (60)	TOD 00B	TOD 10B	TOD 20B (40B)	TOD 30B (50B)	TOE 00	TOE 10 (40)	TOE 20 (50)	TOE 30 (60)	TOD 00A	TOD 10A	APA 00	APA 08
1	TIOC 01	TIOC 11	TIOC 21 (51)	TIOC 31 (61)	TOD 01B	TOD 11B	TOD 21B (41B)	TOD 31B (51B)	TOE 01	TOE 11 (41)	TOE 21 (51)	TOE 31 (61)	TOD 01A	TOD 11A	APA 01	APA 09
2	TIOC 02	TIOC 12	TIOC 22 (52)	TIOC 32 (62)	TOD 02B	TOD 12B	TOD 22B (42B)	TOD 32B (52B)	TOE 02	TOE 12 (42)	TOE 22 (52)	TOE 32 (62)	TOD 02A	TOD 12A	APA 02	APA 10
3	TIOC 03	TIOC 13	TIOC 23 (53)	TIOC 33 (63)	TOD 03B	TOD 13B	TOD 23B (43B)	TOD 33B (53B)	TOE 03	TOE 13 (43)	TOE 23 (53)	TOE 33 (63)	TOD 03A	TOD 13A	APA 03	APA 11
4	TIOC 00	TIOC 10	TIOC 20 (50)	TIOC 40 (70)	TOD 00B	TOD 10B	TOD 20B (40B)	TOD 30B (50B)	TOE 00	TOE 10 (40)	TOE 20 (50)	TOE 30 (60)	TOD 10A	TOD 20A (40A)	APA 04	APA 12
5	TIOC 01	TIOC 11	TIOC 21 (51)	TIOC 41 (71)	TOD 01B	TOD 11B	TOD 21B (41B)	TOD 31B (51B)	TOE 01	TOE 11 (41)	TOE 21 (51)	TOE 31 (61)	TOD 11A	TOD 21A (41A)	APA 05	APA 13
6	TIOC 02	TIOC 12	TIOC 22 (52)	TIOC 42 (72)	TOD 02B	TOD 12B	TOD 22B (42B)	TOD 32B (52B)	TOE 02	TOE 12 (42)	TOE 22 (52)	TOE 32 (62)	TOD 12A	TOD 22A (42A)	APA 06	APA 14
7	TIOC 03	TIOC 13	TIOC 23 (53)	TIOC 43 (73)	TOD 03B	TOD 13B	TOD 23B (43B)	TOD 33B (53B)	TOE 03	TOE 13 (43)	TOE 23 (53)	TOE 33 (63)	TOD 13A	TOD 23A (43A)	APA 07	APA 15
8	TIOC 00	TIOC 10	TIOC 30 (60)	TIOC 40 (70)	TOD 00B	TOD 10B	TOD 20B (40B)	TOD 30B (50B)	TOE 00	TOE 10 (40)	TOE 20 (50)	TOE 30 (60)	TOD 20A (40A)	TOD 30A (50A)	APA 00	APA 08
9	TIOC 01	TIOC 11	TIOC 31 (61)	TIOC 41 (71)	TOD 01B	TOD 11B	TOD 21B (41B)	TOD 31B (51B)	TOE 01	TOE 11 (41)	TOE 21 (51)	TOE 31 (61)	TOD 21A (41A)	TOD 31A (51A)	APA 01	APA 09
10	TIOC 02	TIOC 12	TIOC 32 (62)	TIOC 42 (72)	TOD 02B	TOD 12B	TOD 22B (42B)	TOD 32B (52B)	TOE 02	TOE 12 (42)	TOE 22 (52)	TOE 32 (62)	TOD 22A (42A)	TOD 32A (52A)	APA 02	APA 10
11	TIOC 03	TIOC 13	TIOC 33 (63)	TIOC 43 (73)	TOD 03B	TOD 13B	TOD 23B (43B)	TOD 33B (53B)	TOE 03	TOE 13 (43)	TOE 23 (53)	TOE 33 (63)	TOD 23A (43A)	TOD 33A (53A)	APA 03	APA 11
12	TIOC 00	TIOC 20 (50)	TIOC 30 (60)	TIOC 40 (70)	TOD 00B	TOD 10B	TOD 20B (40B)	TOD 30B (50B)	TOE 00	TOE 10 (40)	TOE 20 (50)	TOE 30 (60)	TOD 30A (50A)	TOD 00A	APA 04	APA 12
13	TIOC 01	TIOC 21 (51)	TIOC 31 (61)	TIOC 41 (71)	TOD 01B	TOD 11B	TOD 21B (41B)	TOD 31B (51B)	TOE 01	TOE 11 (41)	TOE 21 (51)	TOE 31 (61)	TOD 31A (51A)	TOD 01A	APA 05	APA 13
14	TIOC 02	TIOC 22 (52)	TIOC 32 (62)	TIOC 42 (72)	TOD 02B	TOD 12B	TOD 22B (42B)	TOD 32B (52B)	TOE 02	TOE 12 (42)	TOE 22 (52)	TOE 32 (62)	TOD 32A (52A)	TOD 02A	APA 06	APA 14
15	TIOC 03	TIOC 23 (53)	TIOC 33 (63)	TIOC 43 (73)	TOD 03B	TOD 13B	TOD 23B (43B)	TOD 33B (53B)	TOE 03	TOE 13 (43)	TOE 23 (53)	TOE 33 (63)	TOD 33A (53A)	TOD 03A	APA 07	APA 15

Section 19 Window Watchdog Timer (WDTA)

19.1 Features

Channels

This product has two channels of the window watchdog timer.

Table 19.1 Channels of WDTA

Window Watchdog Timer	
Number of Channels	2
Name	WDTA0, WDTA1

Meaning of n

Throughout this section, the individual channels of a window watchdog timer A are identified by the index “n” (n = 0, 1), for example, WDTAnWDTE for the WDTAn enable register.

Register address

All WDTAn register addresses are given as address offsets from the individual base address <WDTAn_base>.

The register base address of each WDTAn is listed in **Table 19.2**.

Table 19.2 Register Base Addresses of WDTAn

WDTAn	<WDTAn_base> Address
WDTA0	FFED 0000_H
WDTA1	FFED 1000_H

Clock supply

The window watchdog timer A provides WDTATCKI and PCLK as the clock inputs.

Table 19.3 Clock Supply of Window Watchdog Timer A

WDTAn	Clock Name for the Unit	Supplied Clock Name
WDTA0	WDTATCKI	WDTACLKI
	PCLK	CLKC_LSB (non-modulated low speed peripheral clock)
WDTA1	WDTATCKI	WDTACLKI
	PCLK	CLKC_LSB (non-modulated low speed peripheral clock)

Interrupts and reset outputs

The interrupts and reset outputs of the WDTAn are listed in **Table 19.4**.

Table 19.4 WDTA Interrupts and Reset Outputs

WDTAn Signal	Function	Connected to
WDTAnTRES	Watchdog reset output signal	ECM
WDTAnTIT	Interval timer interrupt	INTC

19.2 Overview

19.2.1 Functional Overview

The WDTA has the following functions:

- Selection of the operating mode after release from reset using the option byte
  Starting/stopping of the counter after the WDTA is reset, enabling or disabling of 75% interrupt requests, the window-open period, and overflow time can be selected.
- WDTA trigger function
  Writing an activation code to the WDTA trigger register starts WDTA and restarts the counter.
- 75% interrupt request signals
  An interrupt request signal can be generated when the WDTA counter reaches 75% of the overflow interval time (this function can be enabled or disabled by the setting of WDTAnMD.WDTAnWIE).
- Window function
  The period during which writing to the WDTA trigger register is valid (window-open period) can be set. Writing to the WDTA trigger register at a time outside the window-open period causes an error.
- WDTA error detection
  When an error is detected, the WDTAnTRES signal indicates the error to the ECM. For details about the error sources, refer to **Section 19.5.3, Error Detection.**

19.2.2 Block Diagram

Figure 19.1 shows the main components of the WDTA.

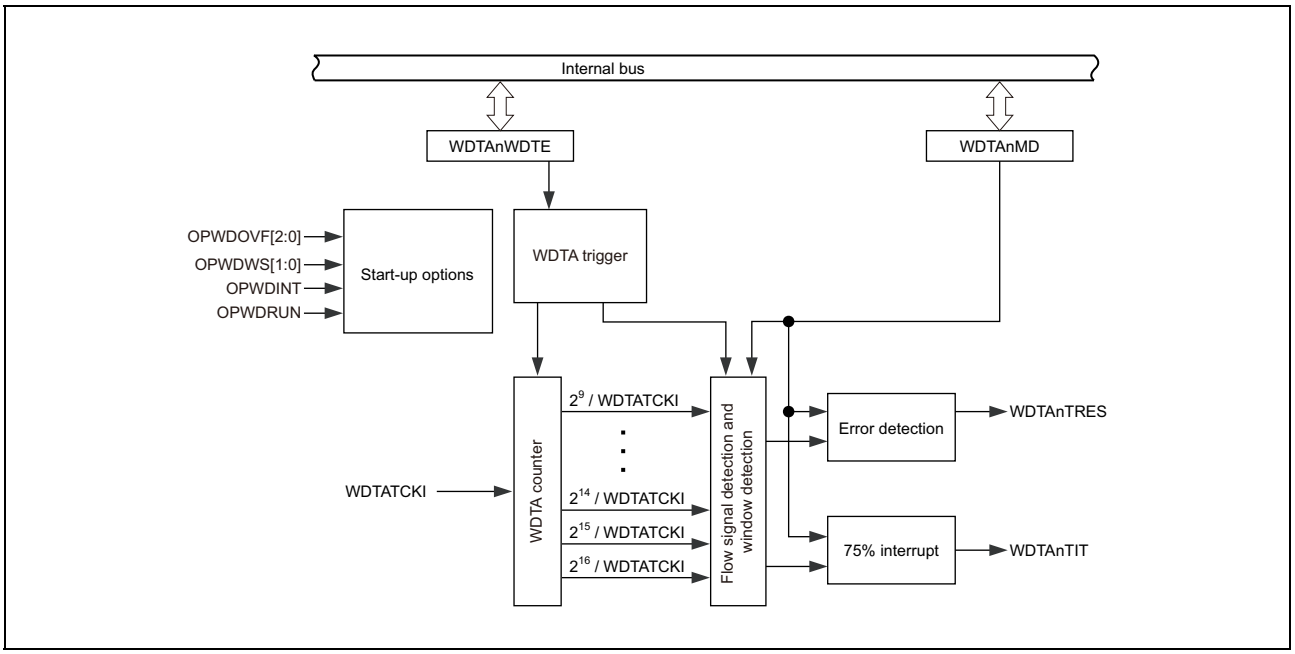


Figure 19.1 Block Diagram of the WDTA

19.3 Registers

The WDTA is controlled and operated by the following registers.

19.3.1 Registers Overview

The following shows the list of registers and the memory address of WDTAn (n = 0, 1).

For the base addresses, see **Table 19.2**.

The actual address can be obtained by adding the offset shown in the table to this base address.

Register Name	Function	R/W	Value after Reset	Access Width			Address
				8	16	32	
WDTAnWDTE	WDTA enable register	R/W	$2C_H$*1	√	—	—	<WDTAn_base> + 0000_H
WDTAnMD	WDTA mode register	R/W	$7F_H$*2	√	—	—	<WDTAn_base> + $000C_H$

Note 1. This value in the WDTA0WDTE register depends on the optional setting value. Refer to **Section 19.3.1.1, WDTAnWDTE — WDTA Enable Register**.

Note 2. This value in the WDTA0MD register depends on the optional setting value. Refer to **Section 19.3.1.2, WDTAnMD — WDTA Mode Register**.

19.3.1.1 WDTAnWDTE — WDTA Enable Register

This register is the WDTA start control and trigger register.

Writing AC_H to this register generates a WDTA trigger and starts or restarts the WDTA counter. See **Section 19.5.2, WDTA Trigger**, for details.

The only value that can be written to this register is AC_H.

Access: This register can be read/written in 8-bit units.

Address: <WDTAn_base> + 0000_H

Value after reset: (1) WDTA0
The value after reset depends on the settings of OPWDRUN. The initial state of OPWDRUN at shipment is OPWDRUN = 0_B. Therefore, the register value after reset of WDTA0 at shipment is $2C_H$.
(2) WDTA1
The register value after reset is also $2C_H$.
See **Section 33.10, Option Bytes** for details of OPWDRUN settings. These registers are initialized by any reset sources.

Bit	7	6	5	4	3	2	1	0
	WDTAnRUN[7:0]							
Value after reset	0	0	1	0	1	1	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 19.5 WDTAnWDTE Register Contents

Bit Position	Bit Name	Function
7 to 0	WDTAnRUN [7:0]	Writing a fixed activation code (AC_H) generates a WDTA trigger to control WDTAn count start and restart. Writing a value other than AC_H generates an error. After WDTAn starts, it cannot be stopped.

The value of bit WDTAnRUN[7] after reset depends on other start-up options as listed below.

Table 19.6 WDTAnRUN[7] Value after Reset

Start-Up Options	
OPWDRUN	**WDTAnRUN[7] Value after Reset**
1	1
0	0

19.3.1.2 WDTAnMD — WDTA Mode Register

This register specifies the overflow interval time, enabling or disabling of the 75% interrupt, and the window-open period.

It can be updated only once after reset release and before the first trigger. The updated value is effective after the next WDTA trigger.

Updating this register after the first WDTA trigger has been generated leads to error detection, but the read value of this register can be written without generating an error.

Access: This register can be read/written in 8-bit units.

Address: <WDTAn_base> + $000C_H$

Value after reset: (1) WDTA0
The value after reset depends on the settings of OPWDOVF2-0, OPWDINT, and OPWDWS1-0. The values of OPWDOVF2-0, OPWDINT, and OPWDWS1-0 are set to 1_B because the initial state of WDTA0 set by the option byte at shipment is the same as that of WDTA1. Therefore, the register value after reset of WDTA0 at shipment is $7F_H$.
(2) WDTA1
The register value after reset is also $7F_H$.

It is initialized by any reset sources.

Bit	7	6	5	4	3	2	1	0
	—	WDTAnOVF[2:0]			WDTAnWIE	—	WDTAnWS[1:0]	
Value after reset	0	1	1	1	1	1	1	1
R/W	R	R/W	R/W	R/W	R/W	R	R/W	R/W

Table 19.7 WDTAnMD Register Contents (1/2)

Bit Position	Bit Name	Function
7	—	Reserved This bit is read as 0. The write value should be 0.
6 to 4	WDTAnOVF[2:0]	These bits select the overflow interval time.

WDTAn OVF2	WDTAn OVF1	WDTAn OVF0	Overflow Interval Time
0	0	0	2^9 / WDTATCKI
0	0	1	2^{10} / WDTATCKI
0	1	0	2^{11} / WDTATCKI
0	1	1	2^{12} / WDTATCKI
1	0	0	2^{13} / WDTATCKI
1	0	1	2^{14} / WDTATCKI
1	1	0	2^{15} / WDTATCKI
1	1	1	2^{16} / WDTATCKI

Table 19.7 WDTAnMD Register Contents (2/2)

Bit Position	Bit Name	Function
3	WDTAnWIE	Enables/disables the 75% interrupt request WDTAnTIT. 0: WDTAnTIT disabled 1: WDTAnTIT enabled
2	—	Reserved This bit is read as 1. The write value should be 1.
1, 0	WDTAnWS[1:0]	These bits select the period over which the window is open.

WDTAnWS1	WDTAnWS0	Window-Open Period
0	0	25%
0	1	50%
1	0	75%
1	1	100%

19.4 Interrupt Sources

The WDTA detects the state of the WDTA counter value or illegal accesses to the WDTA-related registers, and generates an interrupt request. A WDTA interrupt request is described below.

(1) WDTAnTIT (WDTA timer count 75% interrupt request)
An interrupt request signal is generated at 75% of the counter overflow time of the WDTA timer. An interrupt request signal can be enabled or disabled by using the WDTA mode register (WDTAnMD).

19.5 Functional Description

19.5.1 WDTA after Reset Release

19.5.1.1 Start modes

The WDTA provides two modes for the counter start after release from the reset state:

- Software trigger start mode (for WDTA0 and WDTA1)
  The counter value remains 0000_H after reset release.
  The counter is started with the first WDTA trigger.
- Default start mode (only for WDTA0)
  The counter starts automatically after release from the reset state. However, default start mode is disabled in serial programming mode even if OPWDRUN is set to 1_B for the star-up options.

WDTA1 is fixed to software trigger start mode, and cannot be set by the option byte.

19.5.1.2 Start mode selection (only for WDTA0)

The start mode can be selected by the start-up options.

The start mode selection is listed in **Table 19.8**.

Table 19.8 Start Mode Selection

Start-Up Options		
OPWDRUN	**Reset Type**	**Start Mode**
0	Ignored	Software trigger
1		Default

19.5.1.3 WDTA settings after reset release

The WDTA settings are as follows between reset release and the first trigger:

Function	Setting after WDTA0 is Reset	Setting after WDTA1 is Reset
Start mode	Specified by start-up options	Software trigger mode
Overflow interval time	Specified by start-up options	2^{16}/WDTATCKI
75% interrupt mode	Specified by start-up options	75% interrupt enabled
Window-open period	Specified by start-up options	100%

Change WDTA settings

The setting of the WDTA mode register (WDTAnMD) is effective when the first WDTA trigger is generated (writing an activation code to WDTAnWDTE). Change the setting of the WDTAnMD register before a WDTA trigger is generated.

Setting of the WDTA by using WDTAnMD is possible only once. If the value set for WDTAnMD is changed after a WDTA trigger is generated, an error occurs. However, an error does not occur if the same value has been set.

19.5.1.4 Default start mode timing (only for WDTA0)

The default start mode timing and the changes to the WDTA settings are illustrated in **Figure 19.2**.

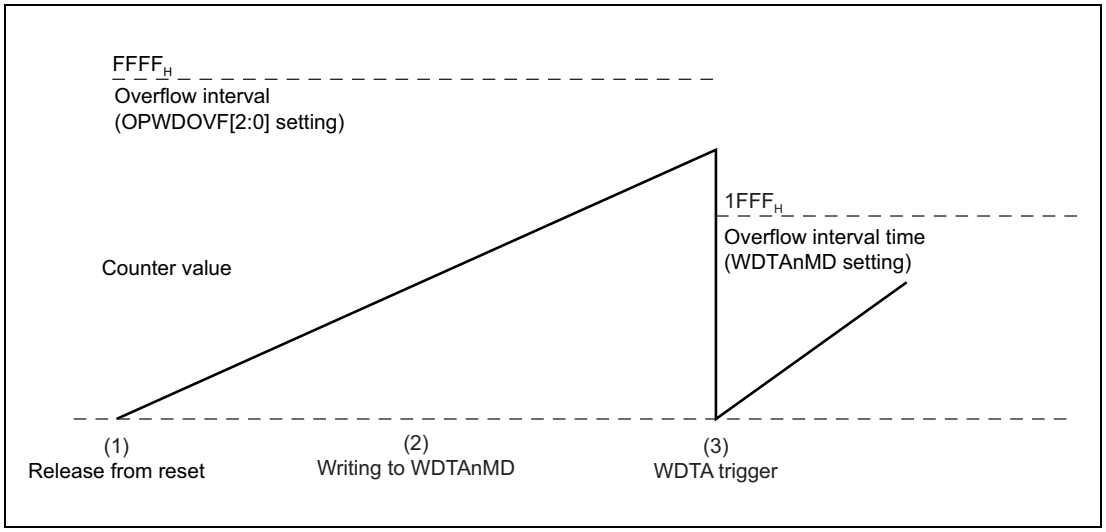


Figure 19.2 Timing Diagram of WDTA Start in Default Start Mode

The timing diagram in **Figure 19.2** shows the following behaviors:

(1) In default start mode, the WDTA counter starts after release from the reset state. The overflow interval time after release from the reset state is set by start-up options.

Example: Overflow interval time after release from the reset state
$= 2^{16}$/WDTATCKI (OPWDOVF[2:0] = 111_B)

(2) WDTAnMD is set before a WDTA trigger is generated. Note, however, that the setting is not applied immediately.

(3) Write to the WDTA trigger register before the WDTA counter overflows.
The WDTAnMD setting is applied due to the WDTA trigger.

Example: Overflow interval time after a WDTA trigger is generated
$= 2^{13}$/WDTATCKI

19.5.1.5 Software trigger start mode timing (common to WDTA0 and WDTA1)

The software trigger start mode timing and the changes to the WDTA settings are illustrated in **Figure 19.3**.

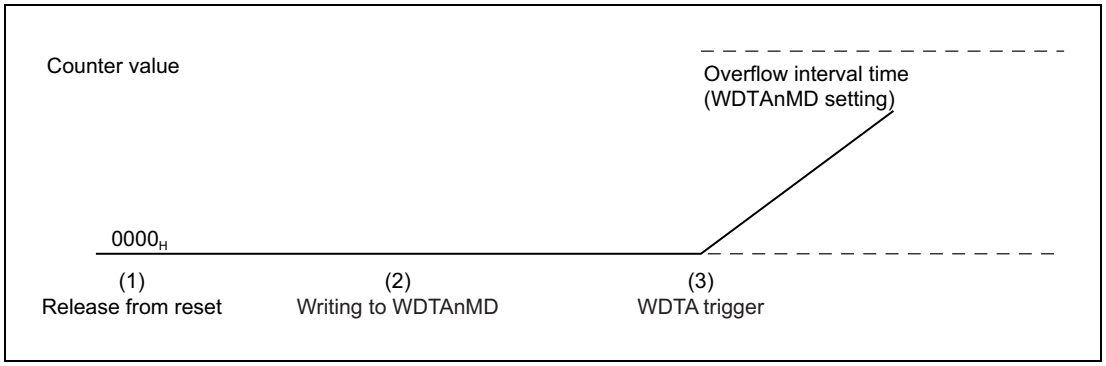


Figure 19.3 Timing Diagram of WDTA Start in Software Trigger Start Mode

The timing diagram in **Figure 19.3** shows the following behaviors:

(1) After release from the reset state, the counter remains 0000_H until the first WDTA trigger.
The overflow interval time is set by using the start-up options, but it does not have any effect.

(2) WDTAnMD is set before the first WDTA trigger. However, the settings are not applied immediately.

(3) The WDTA counter starts at the first WDTA trigger.
The overflow interval time and other settings specified in WDTAnMD are applied.

19.5.2 WDTA Trigger

Writing a special value called an activation code to the WDTA enable register (WDTAnWDTE) leads to generation of a WDTA trigger.

The WDTA trigger has the following functions:

- Starting the WDTA counter in software trigger start mode
- Restarting the WDTA counter
- WDTA mode setting by the WDTAnMD register (only for the first WDTA trigger after release from the reset state)

The WDTA can be triggered by writing a fixed activation code to the trigger register.

Table 19.9 Trigger Register and Activation Code

Type of Activation Code	Trigger Register	Activation Code
Fixed	WDTAnWDTE	AC_H

19.5.3 Error Detection

The WDTA detects an overflow of the WDTA counter and illegal operations as an error.

The conditions for error detection are:

- WDTA counter overflow
- Wrong activation code is written to the WDTA trigger register.
- Writing to the trigger register at the time outside the window-open period
- When an attempt is made to change the setting value of the WDTA mode register WDTAnMD after the first WDTA trigger has been generated
- When updating the setting value of the WDTA mode register WDTAnMD twice before the first WDTA trigger is generated

19.5.4 WDTA Error Mode

When an error is detected, the WDTAnTRES signal indicates the error to the ECM.

Figure 19.4 shows generation of a reset when the counter overflows and default start mode is selected.

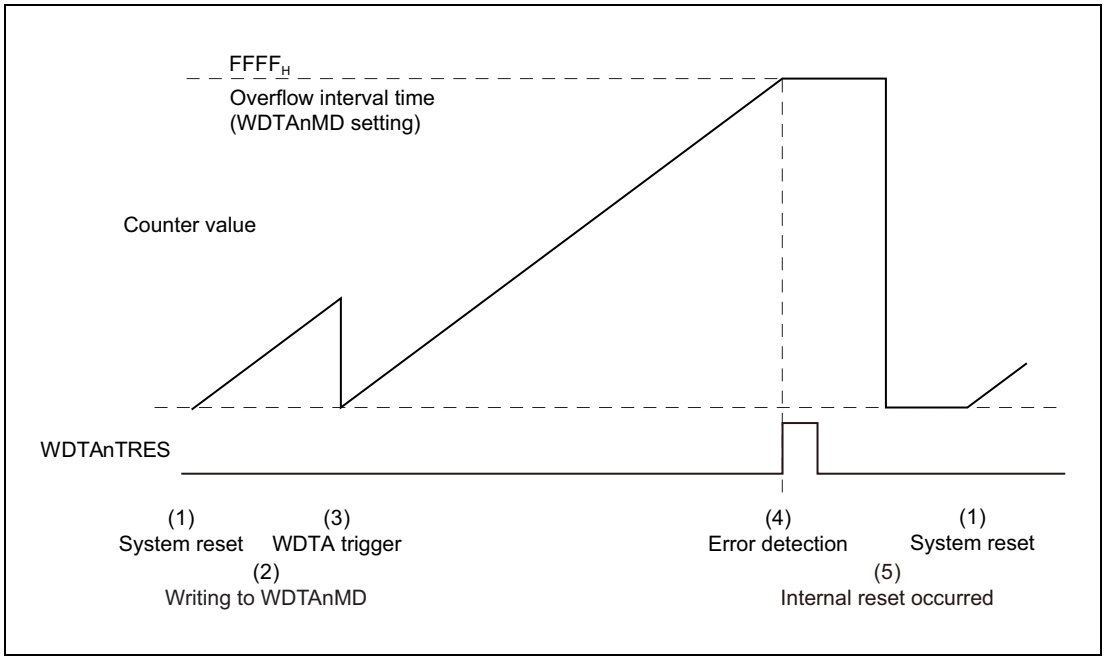


Figure 19.4 Timing Diagram of WDTA Internal Reset Generation

The timing diagram in **Figure 19.4** shows the following behaviors:

(1) In default start mode, the WDTA counter starts after release from the reset state. The overflow interval time after release from the reset state is set by using start-up options.

(2) WDTAnMD is set before a WDTA trigger is generated.
In this case, 2^{16}/WDTATCKI is set for the overflow interval time.

(3) The WDTAnMD setting is applied due to the WDTA trigger.

(4) If the counter overflows, an error is detected and the WDTAnTRES signal indicates the error to the ECM. The counter value remains unchanged until an internal reset occurs.

(5) If an internal reset occurs due to the ECM and other sources, the counter is cleared and stopped until release from the reset state.

19.5.5 75% Interrupt Request Signals

When the WDTA counter reaches 75% of the time set for the overflow interval, an interrupt request WDTAnTIT is generated.

By use of WDTAnMD.WDTAnWIE this function can be enabled or disabled afterwards.

Figure 19.5 shows the 75% interrupt request generation under following conditions:

- Default start mode is selected.
- 75% interrupt request is enabled after the first WDTA trigger is generated.
- WDTA overflow interval time: 2^{16}/WDTATCKI.

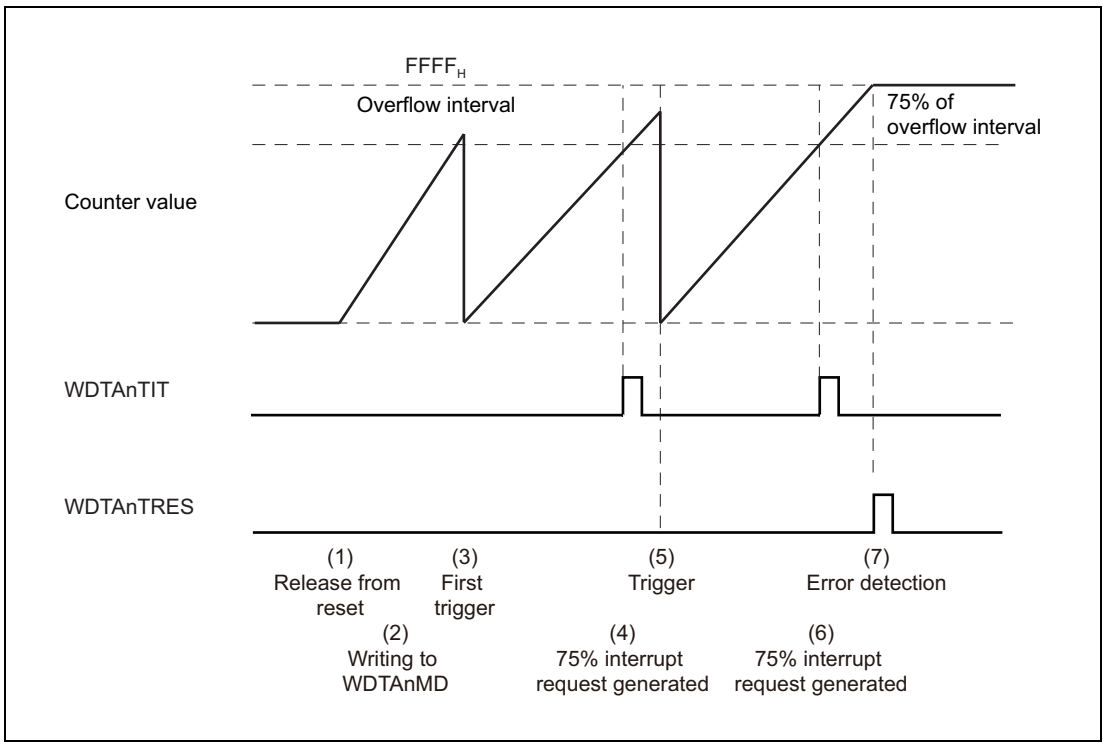


Figure 19.5 Timing Diagram of WDTA 75% Interrupt Request Signals

(1) In default start mode, the WDTA counter starts after release from the reset state. The overflow interval time period after release from the reset state is set by using the start-up options.

(2) WDTAnMD is set before the WDTA trigger is generated. In this figure, 2^{16}/WDTATCKI is set as the overflow interval time period.

(3) The WDTAnMD setting is applied at the WDTA trigger.

(4) When the WDTA counter reaches 75% of the set overflow interval time, an interrupt request WDTAnTIT is generated.

(5) The counter restarts at the WDTA trigger.

(6) When the WDTA counter reaches 75% of the set overflow interval time, an interrupt request WDTAnTIT is generated.

(7) When the counter overflows, an error is detected and the WDTAnTRES signal indicates the error to the ECM. The counter value remains unchanged until an internal reset occurs.

19.5.6 Window Function

The period when a WDTA trigger is valid (window-open period) can be set.
If the window-open period is set to the value less than 100%, an error occurs by the WDTA trigger generated at the time outside the window-open period. The window-open period after release from the reset state is 100%. The period is set to the value set in the WDTAnMD.WDTAnWS[1:0] after the first WDTA trigger is generated.

Figure 19.6 shows window function operations under the following conditions.

- Default start mode is selected.
- The 25% window open period is valid (WDTAnWS[1:0] = 00_B) after the first WDTA trigger.
- WDTA overflow interval time period: 2^{16}/WDTATCKI

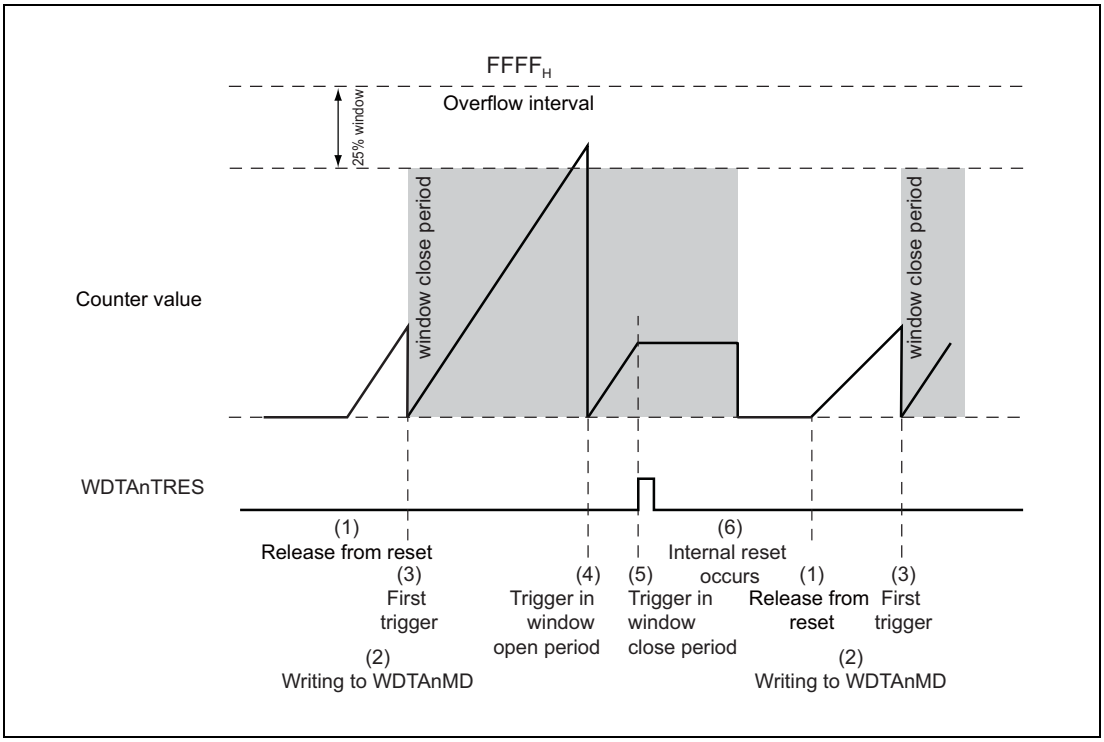


Figure 19.6 Timing Diagram of WDTA Window Function

(1) In default start mode, the WDTA counter starts after release from the reset state. The overflow interval time period after release from the reset state is set by using the start-up options.

(2) WDTAnMD is set before the WDTA trigger is generated. In this figure, 2^{16}/WDTATCKI is set as the overflow interval time period.

(3) The WDTAnMD setting is applied at the WDTA trigger.

(4) The WDTA counter restarts at the WDTA trigger during the window open period.

(5) An error is detected at the WDTA trigger during the window close period, and then the WDTAnTRES signal indicates the error to the ECM. The counter value remains unchanged until an internal reset occurs.

(6) If an internal reset occurs due to the ECM and other sources, the counter is cleared and stopped until release from the reset state.

Section 20 OS Timer (OSTM)

20.1 Functional Overview

The OS timer has the following features.

- Two operating modes
  - Interval timer mode
  - Free-running comparison mode
- Simultaneous starting of multiple channels of the OS timer (timer synchronization of PIC1)
- Interrupt and DTS notification when counting by the counter is started, restarted, or ends.
- Notifying the ECM of errors when an OSTMn (n = 1 only) interrupt occurs.
- Generation of output waveform from the $\overline{\text{ERROROUT\_M}}$ and $\overline{\text{ERROROUT\_C}}$ pins with the OSTM0TTOUT signal of OSTM0 when ECM is in dynamic mode.
- Module standby can be set for each channel (see **Section 12, Standby Controller**, for details).Features of OSTM

20.1.1 Features

Channels

This product has the following number of channels of the OS timer.

Table 20.1 Channels of OS Timer

	OS Timer
Number of channels	3
Name	OSTMn

Meaning of n

Throughout this section, the individual channels of the OS timer are identified by the index “n” (n = 0 to 2), for example, OSTMnTO for the OS timer n output register.

Register address

All OS timer register addresses are given as address offsets from the individual base addresses <OSTMn_base>.

The register base address of each OSTMn is listed in the following table.

Table 20.2 Register Base Addresses

OSTMn	<OSTMn_base> Address
OSTM0	FFEC 0000_H
OSTM1	FFEC 1000_H
OSTM2	FFEC 2000_H

Interrupts

The OS timers can generate the following interrupt requests.

Table 20.3 OSTMn Interrupt Requests

OSTMn Signal	Function	Connected to
OSTM0TINT	OSTM0 interrupt	Interrupt controller INTC
OSTM1TINT	OSTM1 interrupt	Interrupt controller INTC, ECM for error inputs
OSTM2TINT	OSTM2 interrupt	Interrupt controller NTC

20.2 Registers

The OS timers are controlled and operated by the following registers.

20.2.1 Registers Overview

The list of OSTMn (n = 0 to 2) registers and the memory addresses are as follows.

For the base addresses, see **Table 20.2**.

Register Name	Function	R/W	Value after Reset	Access Unit (bit)			Address
				8	16	32	
OSTMnCMP	OSTM compare register	R/W	0000 0000_H	—	—	√	<OSTMn_base> + 00_H
OSTMnCNT	OSTM counter register	R	FFFF $FFFF_H$	—	—	√	<OSTMn_base> + 04_H
OSTMnTO	OSTM output register	R/W	00_H	√	—	—	<OSTMn_base> + 08_H
OSTMnTOE	OSTM output enable register	R/W	00_H	√	—	—	<OSTMn_base> + $0C_H$
OSTMnTE	OSTM count enable status register	R	00_H	√	—	—	<OSTMn_base> + 10_H
OSTMnTS	OSTM count start trigger register	W	00_H	√	—	—	<OSTMn_base> + 14_H
OSTMnTT	OSTM count stop trigger register	W	00_H	√	—	—	<OSTMn_base> + 18_H
OSTMnCTL	OSTM control register	R/W	00_H	√	—	—	<OSTMn_base> + 20_H

20.2.2 Details of OSTM Registers

20.2.2.1 OSTMnCMP — OSTM Compare Register

Depending on the mode of operation, this register holds the start value for the down-counter or the value for comparison with that of the counter.

Access: This register can be read/written in 32-bit units.

Address: <OSTMn_base>

Value after reset: 0000 0000_H
This register is initialized by a reset from any source.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	OSTMnCMP[31:16]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OSTMnCMP[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 20.4 OSTMnCMP Register Contents

Bit Position	Bit Name	Function
31 to 0	OSTMnCMP [31:0]	• In interval timer mode: Start value of the down-counter • In free-running comparison mode: Value for comparison

20.2.2.2 OSTMnCNT — OSTM Counter Register

This register indicates the counter value of the timer.

Access: This register can be read in 32-bit units.

Address: OSTMn_base> + 4_H

Value after reset: FFFF FFFF$_H$
This register is initialized by a reset from any source.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	OSTMnCNT[31:16]															
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OSTMnCNT[15:0]															
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 20.5 OSTMnCNT Register Contents

Bit Position	Bit Name	Function
31 to 0	OSTMnCNT [31:0]	32-bit counter value

Table 20.6 lists the correspondence among the operating mode, counting direction, and start value.

The start value indicates the value to be read after the operating mode is changed.

Table 20.6 Correspondence between Operating Mode, Counting Direction and Start Value

Timer Operating Mode	OSTMnCTL.OSTMnMD1	Counting Direction	Start value
Interval timer mode	0 *1	Down	FFFF FFFF$_H$
Free-running comparison mode	1	Up	0000 0000$_H$

Note 1. Value after reset

20.2.2.3 OSTMnTO — OSTM Output Register

The OSTM output register is used to specify and read the level of an OSTMnTTOUT output signal. The setting of this register is only valid in OSTM0 (n = 0).

Access: This register can be read/written in 8-bit units. Writing can only proceed when the software control mode is selected (OSTMnTOE.OSTMnTOE = 0).

Address: <OSTMn_base> + 8_H

Value after reset: 00_H
This register is initialized by a reset from any source.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	OSTMnTO
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 20.7 OSTMnTO Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When read, the value after reset is read. When writing, write the value after reset.
0	OSTMnTO	This bit specifies/reads the level of an OSTMnTTOUT output signal 0: Low level 1: High level

20.2.2.4 OSTMnTOE — OSTM Output Enable Register

The OSTM output enable register specifies OSTMnTTOUT output mode. The setting of this register is only valid in OSTM0 (n = 0).

Access: This register can be read/written in 8-bit units.

Address: <OSTMn_base> + C_H

Value after reset: 00_H
This register is initialized by a reset from any source.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	OSTMnTOE
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 20.8 OSTMnTOE Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When read, the value after reset is read. When writing, write the value after reset.
0	OSTMnTOE	This bit specifies the OSTMnTTOUT output mode. 0: Software control mode: The level corresponding to the setting of OSTMnTO.OSTMnTO is output to OSTMnTTOUT. 1: Timer-output toggling mode: OSTMnTTOUT output is toggled whenever an OSTMnTINT interrupt request is generated.

20.2.2.5 OSTMnTE — OSTM Count Enable Status Register

This register indicates the state of the counter.

Access: This register can be read in 8-bit units.

Address: <OSTMn_base> + 10_H

Value after reset: 00_H
This register is initialized by a reset from any source.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	OSTMnTE
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 20.9 OSTMnTE Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When read, the value after reset is read.
0	OSTMnTE	This bit indicates whether the counter is operating or stopped. 0: Counter is stopped. 1: Counter is operating. This bit is set to 1 in response to OSTMnTS.OSTMnTS being set to 1, or to the OSTMnTSST signal (when timer synchronization of PIC1 is in use) becoming 1. This bit is reset to 0, if the OSTMnTT.OSTMnTT bit is set to 1.

NOTE

When OSTMnTE = 0, the counter retains its value.

If the counter is restarted,

- It restarts counting down from the value in the OSTMnCMP register in interval timer mode, or
- It restarts counting up from the counter value 0000 0000_H in free-running comparison mode.

20.2.2.6 OSTMnTS — OSTM Count Start Trigger Register

This register starts the counter.

Access: This register can be written in 8-bit units. It is always read as 00_H.

Address: <OSTMn_base> + 14_H

Value after reset: 00_H
This register is initialized by a reset from any source.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	OSTMnTS
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 20.10 OSTMnTS Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When writing, write the value after reset.
0	OSTMnTS	This bit starts the counter. 0: This setting is invalid. 1: Starts the counter and sets OSTMnTE.OSTMnTE = 1. • In interval timer mode, a forced restart is executed if this bit is set while OSTMnTE.OSTMnTE = 1. • In free-running comparison mode, setting this bit is ignored as long as OSTMnTE.OSTMnTE = 1.

20.2.2.7 OSTMnTT — OSTM Count Stop Trigger Register

This register stops the counter.

Access: This register can be written in 8-bit units. It is always read as 00_H.

Address: <OSTMn_base> + 18_H

Value after reset: 00_H
This register is initialized by a reset from any source.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	OSTMnTT
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 20.11 OSTMnTT Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When writing, write the value after reset.
0	OSTMnTT	Stops the counter. 0: This setting is invalid. 1: Stops the counter and clears the OSTMnTE.OSTMnTE bit.

20.2.2.8 OSTMnCTL — OSTM Control Register

This register specifies the operating mode for the counter and controls enabling/disabling of OSTMnTINT interrupt requests when counting starts.

Access: This register can be read/written in 8-bit units. Writing to this register is only possible if the counter is disabled (OSTMnTOE.OSTMnTOE = 0).

Address: <OSTMn_base> + 20_H

Value after reset: 00_H
This register is initialized by a reset from any source.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	OSTMnMD1	OSTMnMD0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 20.12 OSTMnCTL Register Contents

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1	OSTMnMD1	Specifies the operating mode for the counter. 0: Interval timer mode 1: Free-running comparison mode
0	OSTMnMD0	Controls enabling/disabling of OSTMnTINT interrupt requests when counting starts. 0: Disables the interrupts when counting starts. 1: Enables the interrupts when counting starts.

20.3 Functional Description

Each OS timer is a 32-bit timer/counter.

The settings for operating mode specify the direction of counting (up or down) and the generation of interrupt requests.

20.3.1 Block Diagram

The following block diagram shows the main components of OSTMn.

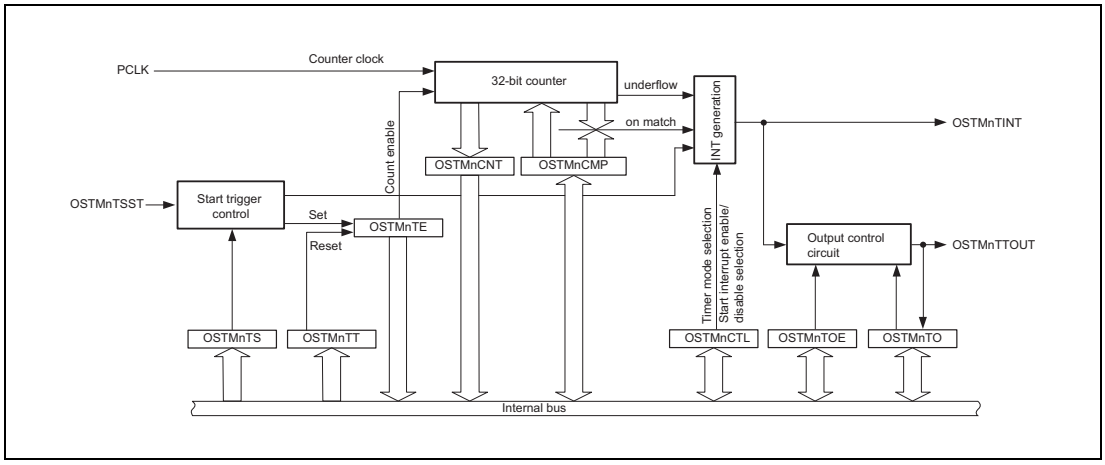


Figure 20.1 Block Diagram of OSTMn

20.3.2 Counter Clock

The counter clock PCLK of OSTMn is CLKC_LSB (non-modulated low speed peripheral clock).

20.3.3 Output Modes (only for OSTM0)

The OSTMn has the following output modes. The mode is selected by the setting of the OSTMnTOE.OSTMnTOE bit.

- Software control mode (the OSTMnTOE.OSTMnTOE bit is 0): The value set in the OSTMnTO.OSTMnTO bit is output to OSTMnTTOUT.
- Timer-output toggling mode (the OSTMnTOE.OSTMnTOE bit is 1): The OSTMnTTOUT output is toggled each time an OSTMnTINT request is generated.

Both output modes are illustrated in the following figure.

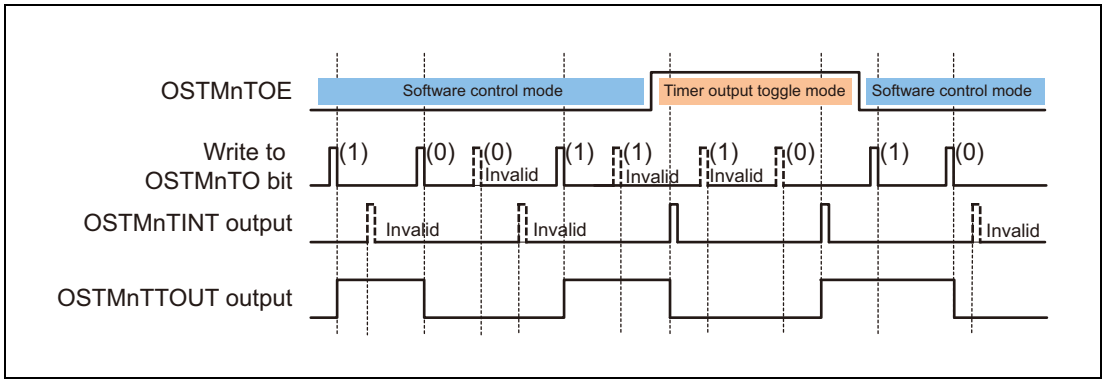


Figure 20.2 Timing Diagram of Output Modes

The above timing diagram shows the following operations.

- In software control mode, the level of the OSTMnTTOUT output changes in accord with the value set in the OSTMnTO.OSTMnTO bit.
- In timer-output toggling mode, the value of the OSTMnTO.OSTMnTO bit and level of the OSTMnTTOUT output are toggled each time an OSTMnTINT interrupt request is generated.

20.3.4 Interrupt Request Generation

An OSTMnTINT interrupt request is generated whenever the counter reaches 0000 0000_H (in interval timer mode) or matches the comparison value (in free-running comparison mode).

An interrupt request can also be generated on starting and restarting of the counter. This is controlled by the OSTMnCTL.OSTMnMD0 bit.

Since OSTMnTINT triggers toggling of the OSTMnTTOUT output in timer output toggling mode (OSTMnTOE.OSTMnTOE is 1), the setting of the OSTMnCTL.OSTMnMD0 bit also affects the output (OSTMnTTOUT).

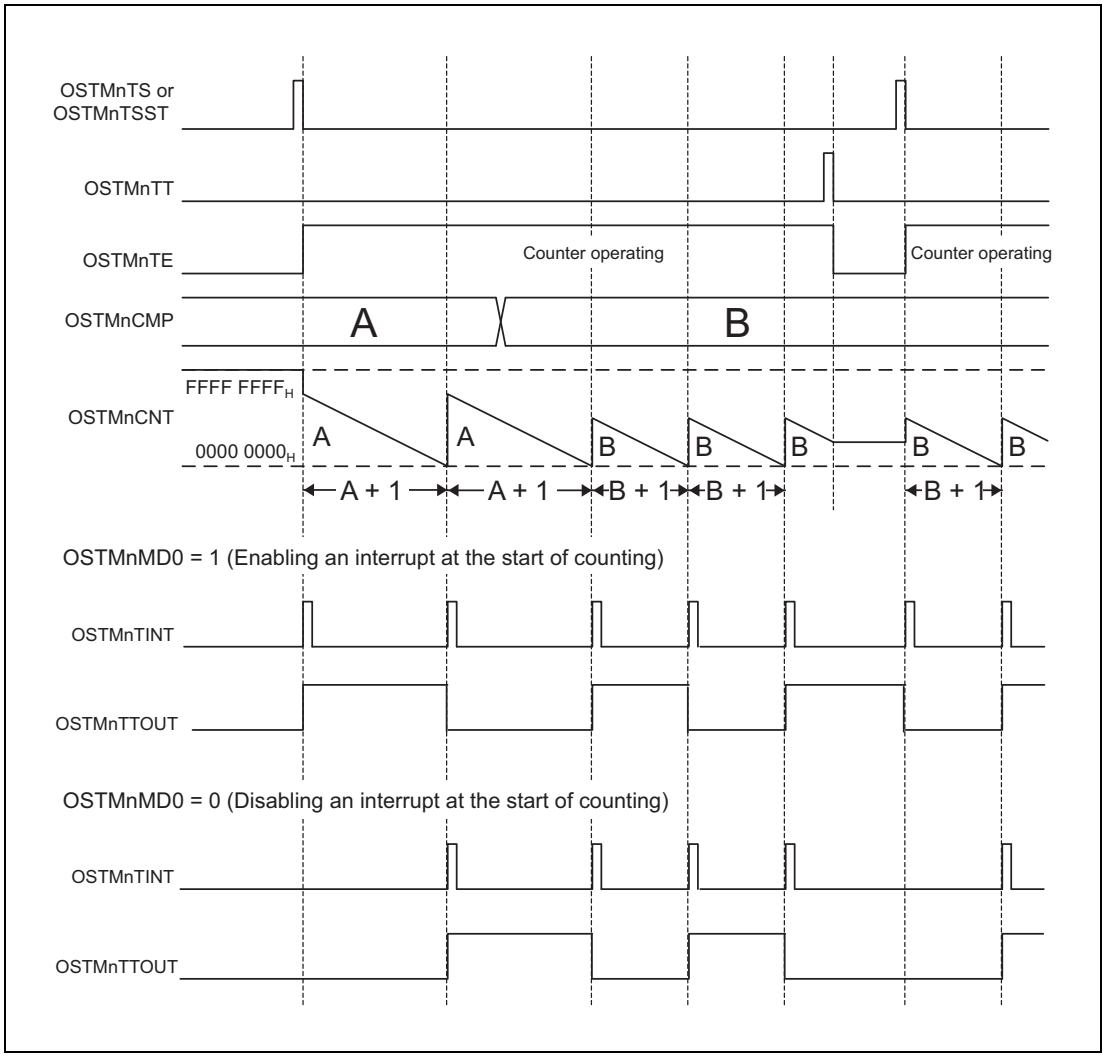


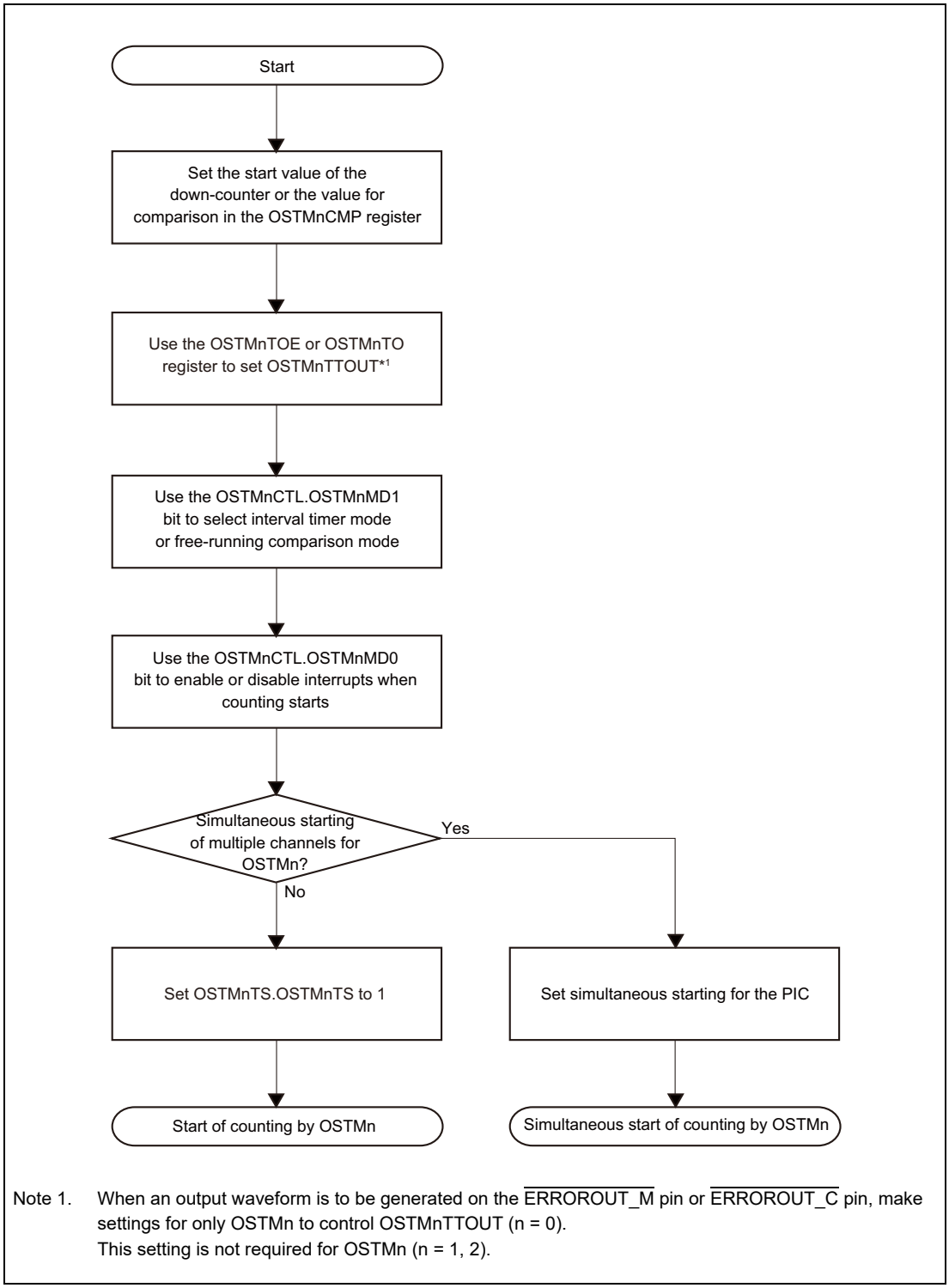
Figure 20.3 Generating an Interrupt when Counting Starts (Interval Timer Mode)

20.3.5 Starting and Stopping the Timer

The OS timer is started and stopped as follows.

Starting the timer

Figure 20.4 shows the flow of starting the counter of OSTMn. For details of the operations of the timer in interval timer mode and free-running timer mode in the given step of the figure, see the descriptions of interval timer mode and free-running comparison mode.



Note 1. When an output waveform is to be generated on the $\overline{\text{ERROROUT\_M}}$ pin or $\overline{\text{ERROROUT\_C}}$ pin, make settings for only OSTMn to control OSTMnTTOUT (n = 0).
This setting is not required for OSTMn (n = 1, 2).

Figure 20.4 Flow of Starting the OS Timer

Stopping the timer

Setting the OSTMnTT.OSTMnTT bit to 1 stops the timer.

This also clears the OSTMnTE.OSTMnTE status flag bit.

Timer Synchronization (PIC1)

The OSTMnTSST signal can be used to start multiple timers at the same time. For detailed connections, refer to **Section 25, Peripheral Interconnection (PIC)**.

20.3.6 Interval Timer Mode

Select the interval timer mode when an OS timer is to be used as a reference timer for generating interrupt requests at a fixed interval.

20.3.6.1 Basic Operation in Interval Timer Mode

In interval timer mode, the timer counts down from the value specified in the OSTMnCMP register. An OSTMnTINT interrupt request is generated when the counter reaches 0000 0000_H.

Select interval timer mode by setting OSTMnCTL.OSTMnMD1 = 0.

New values can be written to the OSTMnCMP register at any time. If it is rewritten during count operation, the counter loads the new OSTMnCMP value when the next 0000 0000_H is reached.

Cycles of OSTMnTINT and OSTMnTTOUT output

The cycles of OSTMnTINT and OSTMnTTOUT output are as follows.

- OSTMnTINT generation cycle = Counter-clock cycle × (OSTMnCMP + 1)
- OSTMnTTOUT output cycle = OSTMnTINT generation period × 2

The following figure shows the basic operation of OSTMn in interval timer mode with counter-start interrupts enabled.

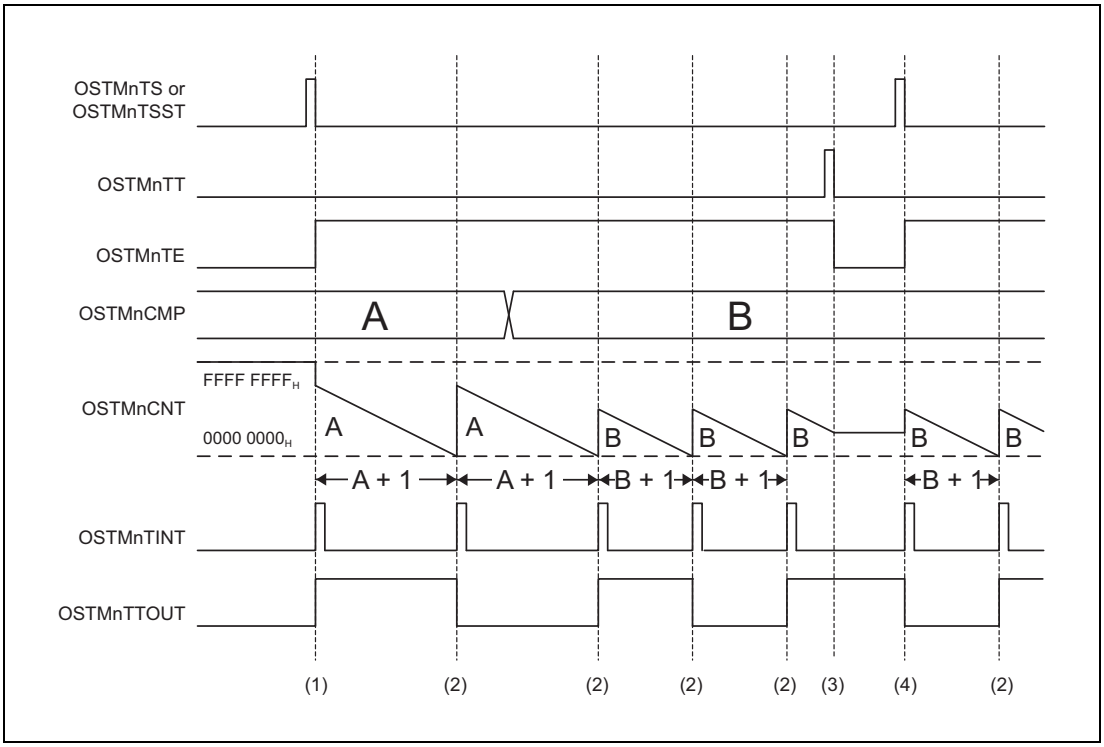


Figure 20.5 Timing Diagram of OSTMn in Interval Timer Mode

The timing diagram above shows the following:

(1) The counter starts counting when OSTMnTS.OSTMnTS = 1 or OSTMnTSST = 1. The OSTMnTE.OSTMnTE bit is set to indicate enabling of the counter.
The counter starts counting-down from the value of OSTMnCMP.
If OSTMnCTL.OSTMnMD0 is 1, OSTMnTINT interrupt requests are generated at the start of

counting and the OSTMnTTOUT signal is toggled. The OSTMnCNT register contains the current value as the counter.

(2) When the counter reaches 0000 0000_H, an OSTMnTINT interrupt request is generated and the OSTMnTTOUT signal is toggled. The counter loads the new start value from OSTMnCMP and continues counting down.

(3) When the counter is stopped (OSTMnTT.OSTMnTT = 1), the OSTMnTE.OSTMnTE bit is cleared to indicate disabling of the counter. The counter retains its current value until it is restarted.

(4) When counting is restarted (OSTMnTS.OSTMnTS = 1, or OSTMnTSST = 1), the counter loads the new start value from OSTMnCMP and starts counting down.

Forced restart

The counter is forcibly restarted by setting OSTMnTS.OSTMnTS = 1 or by transitions of the OSTMnTSST signal from 0 to 1 (when timer synchronization of PIC1 is in use) during counting. The counter loads the initial value from the OSTMnCMP register and continues counting down.

CAUTION

If the OSTMnTS.OSTMnTS bit for a desired channel is set to 1 while simultaneous starting of multiple channels of the timer is in progress, the individual channels are forcibly restarted, so synchronous operation between the channels is not guaranteed. If simultaneous starting of the PIC is used while multiple channels of the timer start individually, the timers set for simultaneous starting are forcibly restarted at the same time.

The counter loads the start value from the OSTMnCMP register and continues counting down.

The following figure shows a timing diagram in interval timer mode, with counter-start interrupts enabled (OSTMnCTL.OSTMnMD0 = 1), and in timer output toggling mode, where OSTMnTTOUT is toggled (OSTMnTOE.OSTMnTOE = 1).

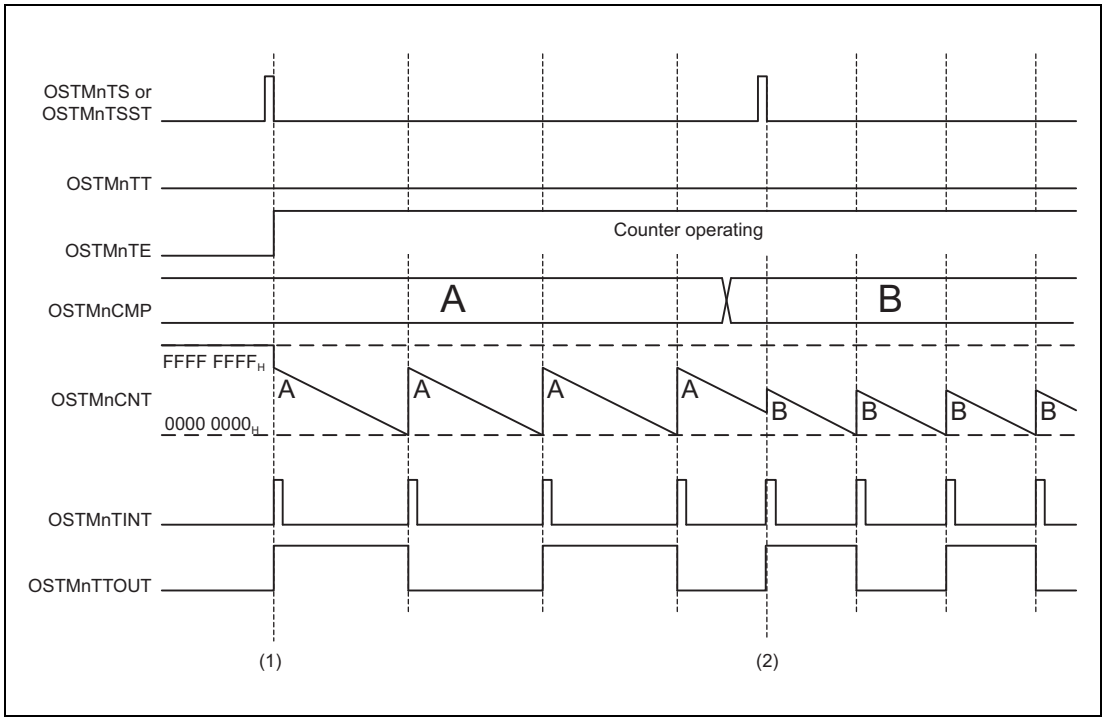


Figure 20.6 Timing Diagram of Forced Restart in Interval Timer Mode

Operations shown in the above timing diagram are as follows.

(1) The counter is started and stopped as described under **Figure 20.5, Timing Diagram of OSTMn in Interval Timer Mode**.

(2) Setting OSTMnTS.OSTMnTS = 1 or OSTMnTSST = 1 restarts the counter while counting is in progress (i.e. while OSTMnTE.OSTMnTE = 1).
The counter immediately restarts counting down, starting with the current value of OSTMnCMP. When OSTMnCTL.OSTMnMD0 = 1, an OSTMnTINT interrupt request is generated when counting starts and the OSTMnTTOUT signal is toggled.

20.3.6.2 Operation when OSTMnCMP = 0000 0000_H

When OSTMnCMP = 0000 0000_H, the OS timer behaves as follows.

- If the counter is enabled, the OSTMnTINT interrupt request will always be set to 1. However, the timer (OSTMnTTOUT) output signal can still be used. Timer (OSTMnTTOUT) output using timer output toggling mode results in OSTMnTTOUT being toggled on every cycle of the counter clock.

The following figure shows operations of the OSTMn when OSTMnCMP = 0000 0000_H, the counter start interrupt is enabled (OSTMnCTL.OSTMnMD0 = 1), and the timer is in timer output toggling mode, where OSTMnTTOUT is toggled (OSTMnTOE.OSTMnTOE = 1).

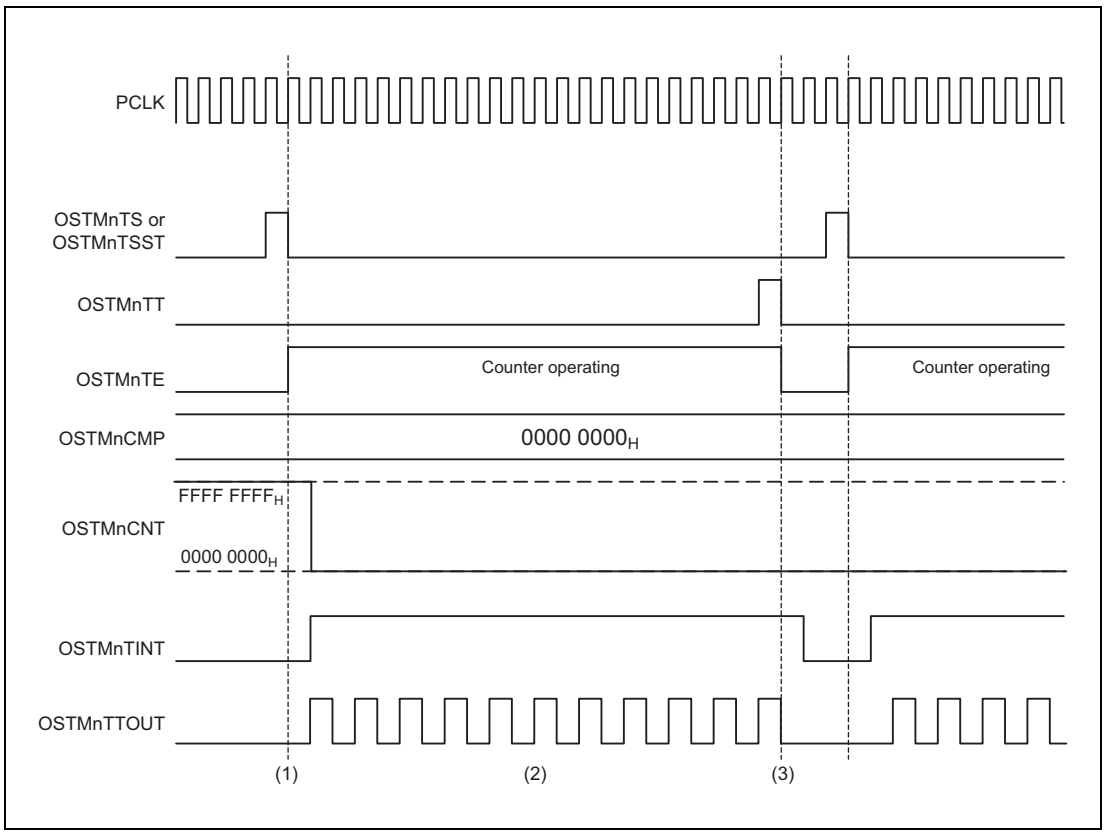


Figure 20.7 Timing Diagram when OSTMnCMP = 0000 0000_H in Interval Timer Mode

The timing diagram above shows the following operations:

(1) The counter is reloaded with the value in OSTMnCMP as soon as it starts counting, so the value 0000 0000_H is retained in OSTMnCMP.

(2) While the OSTMnTINT interrupt request signal is continuously asserted, OSTMnTTOUT is toggled (in **Figure 20.7**, OSTMnTINT is fixed to the high level because PCLK is selected as the counter clock).

(3) After the counter stops, the OSTMnTINT interrupt request signal is deasserted and the output level of the OSTMnTTOUT signal is maintained.

When an interrupt is prohibited when the counter starts, 1 clock cycle of the counter clock is not generated at the start timing of the counter.

20.3.7 Free-Running Comparison Mode

20.3.7.1 Basic Operation in Free-Running Comparison Mode

In free-running comparison mode, the counter counts up from 0000 0000_H to FFFF $FFFF_H$. An OSTMnTINT interrupt request is output when the current value of the counter matches the value of the OSTMnCMP register. The free-running comparison mode is selected by setting the OSTMnCTL.OSTMnMD1 bit to 1.

New values can be written to the OSTMnCMP register at any time.

The following figure shows the basic operation of OSTMn in free-running compare mode with the start of counting enabled (OSTMnCTL.OSTMnMD0 = 1) and with the timer in timer output toggling mode, where OSTMnTTOUT is toggled (OSTMnTOE.OSTMnTOE = 1).

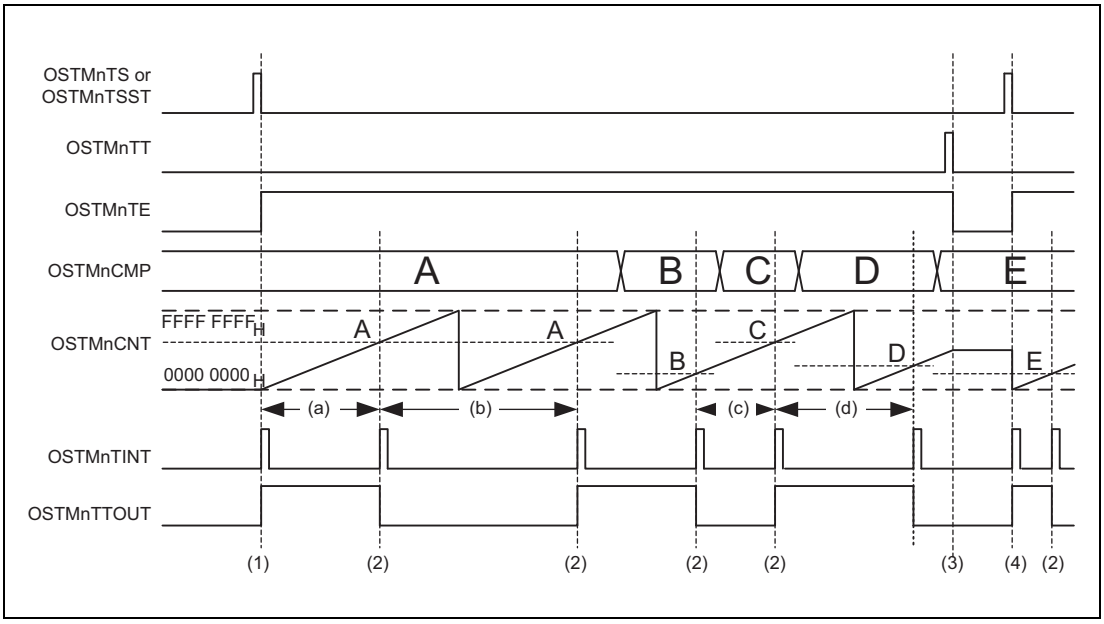


Figure 20.8 Timing Diagram of OSTMn in Free-Running Comparison Mode

The timing diagram above shows the following:

(1) The counter starts counting when OSTMnTS.OSTMnTS = 1, or when OSTMnTSST = 1. The OSTMnTE.OSTMnTE bit is set to indicate enabling of the counter. The counter counts up from 0000 0000_H to FFFF $FFFF_H$. The OSTMnCNT register is the counter, so it contains the current value. When the OSTMnCTL.OSTMnMD0 set to 1, the interrupt request OSTMnTINT is generated when the counting starts.

(2) When the current counter value matches the value in the OSTMnCMP register, an OSTMnTINT interrupt request is generated and the OSTMnTTOUT signal is toggled.

(3) When the counter is stopped (OSTMnTT.OSTMnTT = 1), the OSTMnTE.OSTMnTE bit is cleared to indicate disabling of the counter.
The counter retains its current value until it is restarted.

(4) Counting by the counter restarts from 0000 0000_H when OSTMnTS.OSTMnTS = 1, or when OSTMnTSST = 1.

OSTMnTINT Period

The OSTMnTINT generation period is different at the start of counting and depends on the old and new comparison values if OSTMnCMP is rewritten during operation.

Table 20.13 Timing of OSTMnTINT Generation

Old value for comparison	New value for comparison	Counter value at time of rewriting	Period of OSTMnTINT Generation	Label in timing diagram
Counter starts			(A + 1) × counter clock period	(a)
A	A	No rewriting	(FFFF FFFF$_H$ + 1) × counter clock period	(b)
B	C > B	B < counter value < C	(C - B) × counter clock period	(c)
C	D < C	Counter value > D,C	(FFFF FFFF$_H$ - C + D + 1) × counter clock period	(d)

Forced restart

Forced restarting does not proceed during counting even if the OSTMnTS.OSTMnTS bit is set, or if OSTMnTSST is high when the simultaneous start trigger is in use. The counter ignores the attempted setting and continues counting.

20.3.7.2 Operation when OSTMnCMP = 0000 0000$_H$

The following figure shows the operation of OSTMn when OSTMnCMP = 0000 0000$_H$, counter-start interrupts are enabled (OSTMnCTL.OSTMnMD0 = 1) and the timer is in timer output toggling mode, where OSTMnTTOUT is toggled (OSTMnTOE.OSTMnTOE = 1).

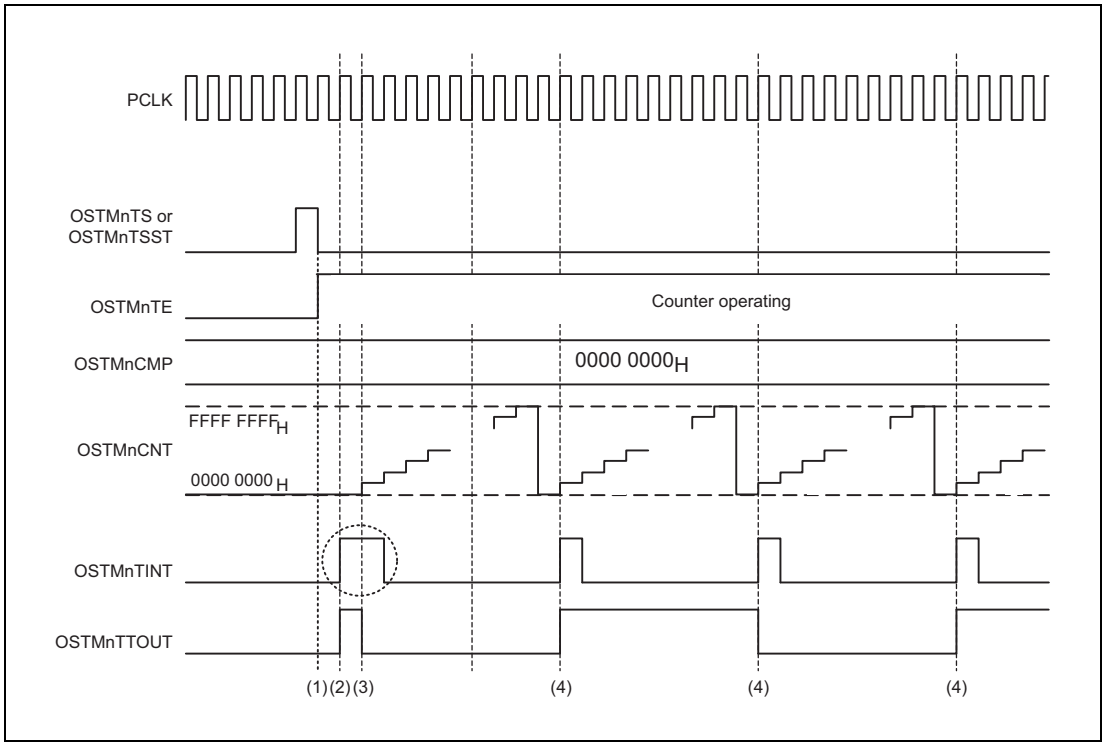


Figure 20.9 Timing Diagram when OSTMnCMP = 0000 0000$_H$ in Free-Running Comparison Mode

The timing diagram above shows the following operations.

(1) Once the counter starts, it counts up from 0000 0000$_H$ to FFFF FFFF$_H$.

(2) An OSTMnTINT interrupt request is generated when counting starts and the OSTMnTTOUT signal is toggled.

(3) If the current counter value matches OSTMnCMP, the interrupt request OSTMnTINT is generated.
If OSTMnCMP = 0000 0000$_H$ in the above case, OSTMnTINT is generated over two clock cycles and the OSTMnTTOUT signal is toggled.

(4) For every (FFFF FFFF$_H$ + 1) clock cycles the OSTMnTINT interrupt request signal is asserted and the OSTMnTTOUT signal is toggled.

When an interrupt is prohibited when the counter starts, 1 clock cycle of the counter clock is not generated at the start timing of the counter.

Section 21 Advanced Timer Unit IV (ATU-IV)

21.1 Overview

The ATU-IV has the following features.

- Processing of up to 166 pulse input/output signals
- 297 interrupt sources can be generated. This enables to activate the general direct memory access controller (DMAC), and interrupt processing by the CPU.
- Equipped with a function to output pulses to A/D and DFE (64 pulses to ADC (16 for timer C, 40 for timer D, and 8 for timer G), 48 pulses to ΔΣAD (40 for timer D and 8 for timer G), and 40 pulses to DFE (40 for timer D))
- On-chip 4-channel prescaler provided, which generates four types of clocks by dividing on-chip peripheral clock (PCLK) by 1/1 to 1/1024
- Each channel for a timer can select a count source from among four divided clocks generated by prescaler, two external clocks, and angle clock generated by timer B.
- Pin output values of timer C, D, and E can be selected by the RHSB cross bar and then output as data to the RHSB.
- Counter values of timer D and G can be selected by the APA input selector and then output as reference data to the APA. Timer D counter compare match, counter underflow, pin output signals of timer D, and timer G counter compare match can be output as event input signals to the APA.
- Timer A has a 32-bit free-run counter and seven 32-bit input capture registers. Features are shown below.
  - Detection by rising edges, falling edges, or both edges
  - Activating the DMAC at the capture timing
  - Noise canceling function for each external input pin with maximum length of 0.21 s or noise cancelling function using the angle clock
  - The level of each external input pin can be read or captured.
  - A capture interrupt request and a counter overflow interrupt request can be generated.
  - Settings for noise cancellation mode can be made in subblock/channel units.
  - Signals after noise cancellation can be sent to external input pins of timer F.
- Timer B consists of three subblocks: an edge-interval measuring block, frequency-multiplied clock generator, and frequency-multiplied clock signal corrector.
- The edge-interval measuring block is provided with a 32-bit input edge-interval measuring timer, three output compare and input capture registers, six edge-interval time measured value recording registers, seven edge-interval time measured value history registers, 8-bit event counter, and output compare register. This provides the following operations:
  - Capture by edges of external event input (rising edge, falling edge, or both edges are selectable)
  - Capture by event compare match of external event input
  - Automatic event counter clear by event compare match
  - Capture interrupt and compare match interrupt (edge-interval compare match, event compare match)

  - Captures event counter values individually by detection of seven external event inputs from timer A.

- The frequency-multiplied clock generator is provided with 24-bit reload counter, reload register, 20-bit multiplied clock counter, 26-bit multiplied clock counter, capture register and output compare register. This provides the following operations:
  - Reloadable counting of values captured by edge-interval measuring block with arbitrary number (1 to 4095)
  - Reloadable down-count values can be modified by event compare match of the edge-interval measuring block.
  - Internal clock generated by the underflow of the reload counter can be used as input of 20-bit multiplied clock counter and 26-bit multiplied clock counter.
  - The 20-bit multiplied clock counter value can be captured by an edge input of external input event.
  - An interrupt request for compare match between 20-bit multiplied clock counter and output compare register and an interrupt request for comparison between 26-bit multiplied clock counter and capture register can be generated.
- Frequency-multiplied clock signal corrector is provided with 20-bit correcting event counter, 20-bit correcting multiplied clock counter, multiplied-and-corrected clock generating counter, and correcting counter clearing register. This provides the following operations:
  - Frequency-multiplied correcting clock that serves as the count source for other timers can be generated based on the read count in frequency-multiplied clock generator.
  - The free-running counter of timer D can be cleared by compare match (enabling or disabling this function selectable) between multiplied-and-corrected clock generating counter and correcting counter clearing register.
  - The correcting event counter can be automatically cleared at the event compare match timing due to an external input event of the edge-interval measuring block.
  - Additionally corrected multiplied-and-corrected clock 2 (AGCKM2) can be output to the multiplied-and-corrected clock (AGCKM).
- Timer C consists of eight subblocks that have the same functions. Each subblock consists of four channels. Each subblock is provided with a 32-bit free-running counter, four 32-bit general registers, four 32-bit compare match registers, and a 32-bit upper-limit value setting register. This provides the following operations:
  - Input capture or output compare is selectable
  - Detection edge for input capture is selectable from among rising edge, falling edge, or both edges.
  - When timer C is used as input capture, capturing is possible at the timing of event output 1, 2A, and 2B from timer A.
  - Noise canceling function with maximum length of 0.21 s or noise cancelling function using the angle clock for each external input pin
  - Selectable noise canceling mode per sub-block or channel
  - 1, 0, or toggle can be output by a compare match.
  - Three PWM waveforms can be output for each subblock in PWM mode.
  - Input capture/compare match interrupt and overflow interrupt can be generated. Input capture/compare match interrupts activate DMAC.

  - One-shot pulse mode can be output.
  - Upper-limit value setting function for 32-bit counters

- Timer D consists of 10 subblocks that have the same function Each subblock is provided with two 32-bit free running counters, an offset base register, an output value register, and an output setting register, and four channels. Each channel is provided with two 32-bit output compare registers, two range comparison value setting registers, two input capture registers, a 32-bit down-counter for outputting one-shot pulse, and two 32-bit upper-limit value setting registers. This provides the following operations:
  - Enables to start down-counter by software. One-shot pulse can also be generated.
  - Compare match between compare match register and general register can be used as the start trigger for down-counter. One-shot pulse with offset can also be generated.
  - Compare match in compare match register can stop down-counter and forcibly cut off one-shot pulse output.
  - Compare match between compare match register and general register can be output.
  - Compare match range can be set by the range comparison value setting register (range compare function).
  - Compare match in compare match register can be used as a trigger and the count number can be captured by input capture register.
  - Free run value can be captured triggered by timer A
  - Provided with counter clearing function from timer B
  - Generation of 80 compare match interrupts, 20 counter overflow interrupts, and 40 underflow interrupts. The 24 compare match interrupts and 20 underflow interrupts correspond to DMAC activation.
  - Output pulses that show interrupts of compare match A or compare match B for A/D activation. (20 for each)
  - Output waveform can be inverted.
  - The value set in the output value register can be output from each pin by the output setting register.
  - Timer D counter compare match, counter underflow, and pin output signal of timer D can be output as event output to the APA (subblock 0).
  - Free-running counter value can be output as reference data to the APA (subblock 0).
  - Upper-limit value setting function for 32-bit counters
- Timer E consists of seven subblocks that have the same function. Each subblock consists of four channels. Each channel is provided with 24-bit free running counter, duty cycle setting register, cycle setting register, duty cycle reload register, and cycle reload register. This provides the following operations:
  - PWM output with programmable cycle time and duty cycle ranging from 0 to 100%
  - Switchable between on-state and off-state duty modes
  - Values in duty cycle reload register/cycle reload register can be transferred to duty cycle setting register/cycle setting register at every cycle.
  - Writing $000\ 0000_H$ to the 24 higher-order bits of the counter can forcibly end PWM cycle and start new PWM cycle.
  - PWM output can be blocked by shutoff input.

  - 28 interrupt requests can be output at each cycle for cycle match interrupt requests and duty match interrupt request (Channel 0 of each subblock supports for DMAC activation by cycle match)

- Timer F consists of 16 subblocks. Each subblock is provided with two 32-bit counters, a 16-bit counter, three (for all subblocks) 32-bit general registers, and a 16-bit general register. This provides the following operations:
  - Detection by rising edge, falling edge or both edges
  - TIA00 to TIA06 signals of timer A after noise cancellation can be used as external input signals.
  - Noise canceling function for each external pin with maximum length of 0.21 s or noise cancelling function using the angle clock.
  - Selectable noise canceling mode per sub-block or channel
  - Seven operation modes: edge counting in a specified period, effective edge interval counting, measurement of time during high/low input, measurement of PWM input waveform timing, rotation speed/pulse measurement, up/down event count, and four-time multiplication event count.
  - Activates DMAC by input capture interrupt request
  - Overflow interrupt generation
  - A compare match interrupt request can be generated in PWM input waveform measurement mode or rotation speed/pulse measurement mode.
  - Settings for noise cancellation mode can be made in channel units.
- Timer G consists of 10 subblocks that have the same function. Each channel is provided with a 32-bit free-run counter and output compare register. This provides the following operations:
  - A compare match trigger event can be output and used as A/D activation/interrupt request trigger and event output to the APA.
  - Activates DMAC by compare match interrupt request
  - Free-running counter value can be output as reference data to the APA (subblock 0).
- Timer H consists of a pair of 16-bit counter and 16-bit compare match register, and a 32-bit counter. This provides the following operations:
  - Measurement of time ranging from 1 to 2^{26} times of peripheral internal clock (PCLK) using a 16-bit counter and a 16-bit compare match register. Can be output as compare match interrupt.
  - Equipped with a 32-bit counter to count the compare match occurrence.
- Timer J consists of 6 subblocks that have the same function. Each channel is provided with a 32-bit counter, output compare register, and nine-stage FIFO register. This provides the following operations:
  - Detection by rising edge, falling edge or both edges
  - Noise canceling function with maximum length of 0.21 s or noise cancelling function using the angle clock for each external input pin.
  - Selectable noise canceling mode per sub-block
  - Capture the counter value in FIFO register (edge input interval) when detecting edges on external input pin.
  - Activates DMAC at the timing when FIFO is full.
  - Controls FIFO's effective capture time using compare match register
  - FIFO full interrupt, counter overflow, and FIFO overflow interrupt can be generated.

21.1.1 Configuration of ATU-IV

ATU-IV consists of nine timer blocks (timer A to timer J), prescalers, and a controller. The timer blocks have different functions and each can operate independently; timer blocks can also be linked via the clock bus. Each timer block consists of one or more timer subblocks and each subblock has one or more channels. (See **Table 21.1, Block Configuration for ATU-IV**, and **Section 21.1.2, ATU-IV Registers**).

Table 21.1 Block Configuration for ATU-IV

Modules	Blocks	Subblock	Remarks
ATU-IV			
	Common controller unit		
	Prescaler		Total number of channels: 4
		Channel 0	
		:	
		Channel 3	
	Timer A		Total number of channels: 7
		Channel 0	
		:	
		Channel 6	
	Timer B		Total number of channels: 1
	Timer C		Total number of channels: 32 Number of subblock: 8 Number of channel/subblock: 4
		Timer C0	
		Channel 0	
		:	
		Channel 3	
	Timer C1 to Timer C7 (Details are omitted)		
	Timer D		Total number of channels: 40 Number of subblock: 10 Number of channel/subblock: 4
		Timer D0	
		Channel 0	
		:	
		Channel 3	
	Timer D1 to Timer D9 (Details are omitted)		
	Timer E		Total number of channels: 28 Number of subblock: 7 Number of channel/subblock: 4
		Timer E0	
		Channel 0	
		:	
		Channel 3	
	Timer E1 to Timer E6 (Details are omitted)		
	Timer F		Total number of channels: 16 Number of subblocks: 16 Number of channel/subblock: 1
		Timer F0	
		:	
		Timer F15	
	Timer G		Total number of channels: 10 Number of subblocks: 10 Number of channel/subblock: 1
		Timer G0	
		:	
		Timer G9	
	Timer H		Total number of channels: 1
	Timer J		Total number of channels: 6 Number of subblocks: 6 Number of channel/subblock: 1
		Timer J0	
		:	
		Timer J1 to J5	

21.1.2 ATU-IV Registers

Addresses of the ATU-III registers are shown below. To access the registers, the following procedure should be followed.

1. When writing to reserved bits, always write 0.
2. There are restrictions on register write access size for some registers. Be careful when making a write access with a bit width smaller than the register bit width. For details, see the description of each register.

The following shows address allocation of ATU-IV registers.

Common Controller Registers (FFE6 0000_H to FFE6 $007F_H$)

Address	b31 +3 b24	b23 +2 b16	b15 +1 b8	b7 +0 b0	Page
FFE6 0000_H	NCMR	CBCNT	ATUENR		ATU-IV
FFE6 0004_H	(Prohibited area)				
FFE6 0008_H	(Prohibited area)				
FFE6 $000C_H$ to FFE6 $007C_H$	(Prohibited area)				

Prescaler Registers (FFE6 0080_H to FFE6 $00BF_H$)

Address	b31 +3 b24	b23 +2 b16	b15 +1 b8	b7 +0 b0	Page
FFE6 0080_H	(Prohibited area)		PSCR0		PSC
FFE6 0084_H	(Prohibited area)		PSCR1		
FFE6 0088_H	(Prohibited area)		PSCR2		
FFE6 $008C_H$	(Prohibited area)		PSCR3		
FFE6 0090_H	(Prohibited area)				
FFE6 0094_H	(Prohibited area)				
FFE6 0098_H to FFE6 $00BC_H$	(Prohibited area)				PSC

DMA/AD Requests Automatic Switching Registers (FFE6 $00C0_H$ to FFE6 $00FF_H$)

Address	b31 +3 b24	b23 +2 b16	b15 +1 b8	b7 +0 b0	Page
FFE6 $00C0_H$	(Prohibited area)			TRGSRDMA0	
FFE6 $00C4_H$	(Prohibited area)	TRGSELDMA01	(Prohibited area)	TRGSELDMA00	
FFE6 $00C8_H$	(Prohibited area)			TRGSELAD	
FFE6 $00CC_H$	(Prohibited area)				
FFE6 $00D0_H$	(Prohibited area)			TRGSRDMA1	
FFE6 $00D4_H$	(Prohibited area)	TRGSELDMA11	(Prohibited area)	TRGSELDMA10	
FFE6 $00D8_H$ to FFE6 $00F8_H$	(Prohibited area)				
FFE6 $00FC_H$	(Prohibited area)				

Timer A Registers (FFE6 0200$_H$ to FFE6 03FF$_H$) (1/2)

Address	b31 +3 b24	b23 +2 b16	b15 +1 b8	b7 +0 b0	Page
FFE6 0200$_H$	(Prohibited area)	TCR2A	(Prohibited area)	TCR1A	
FFE6 0204$_H$	(Prohibited area)	TCR4A	(Prohibited area)	TCR3A	
FFE6 0208$_H$	(Prohibited area)	TSCRA	(Prohibited area)	TSRA	
FFE6 020C$_H$	(Prohibited area)	NCMCR2A	(Prohibited area)	NCMCR1A	
FFE6 0210$_H$	(Prohibited area)		TIOR1A		
FFE6 0214$_H$	TIOR2A				
FFE6 0218$_H$	(Prohibited area)	TILCRA	(Prohibited area)	TILRA	
FFE6 021C$_H$	(Prohibited area)				
FFE6 0220$_H$	TCNTA				
FFE6 0224$_H$ to FFE6 023C$_H$	(Prohibited area)				
FFE6 0240$_H$	ICRA0				
FFE6 0244$_H$	NCRA0		NCNTA0		
FFE6 0248$_H$ to FFE6 025C$_H$	(Prohibited area)				
FFE6 0260$_H$	ICRA1				
FFE6 0264$_H$	NCRA1		NCNTA1		
FFE6 0268$_H$ to FFE6 027C$_H$	(Prohibited area)				
FFE6 0280$_H$	ICRA2				
FFE6 0284$_H$	NCRA2		NCNTA2		
FFE6 0288$_H$ to FFE6 029C$_H$	(Prohibited area)				
FFE6 02A0$_H$	ICRA3				
FFE6 02A4$_H$	NCRA3		NCNTA3		
FFE6 02A8$_H$ to FFE6 02BC$_H$	(Prohibited area)				
FFE6 02C0$_H$	ICRA4				
FFE6 02C4$_H$	NCRA4		NCNTA4		
FFE6 02C8$_H$ to FFE6 02DC$_H$	(Prohibited area)				
FFE6 02E0$_H$	ICRA5				
FFE6 02E4$_H$	NCRA5		NCNTA5		
FFE6 02E8$_H$ to FFE6 02FC$_H$	(Prohibited area)				
FFE6 0300$_H$	ICRA6				
FFE6 0304$_H$	NCRA6		NCNTA6		

Timer A Registers (FFE6 0200_H to FFE6 $03FF_H$) (2/2)

Address	b31 +3 b24	b23 +2 b16	b15 +1 b8	b7 +0 b0	Page
FFE6 0308_H	(Prohibited area)				
to					
FFE6 $031C_H$					
FFE6 0320_H	(Prohibited area)				
to					
FFE6 $03DC_H$					
FFE6 $03E0_H$	(Prohibited area)				
to					
FFE6 $03FC_H$					

Timer B Registers (FFE6 0400$_H$ to FFE6 05FF$_H$) (1/2)

Address	b31 +3 b24	b23 +2 b16	b15 +1 b8	b7 +0 b0	Page
FFE6 0400$_H$	(Prohibited area)	TIORB	(Prohibited area)	TCRB	
FFE6 0404$_H$	TSCRB		TSRB		
FFE6 0408$_H$	(Prohibited area)			TICRB	
FFE6 040C$_H$ to FFE6 0478$_H$	(Prohibited area)				
FFE6 047C$_H$	(Prohibited area)				
FFE6 0480$_H$	TCNTB0				
FFE6 0484$_H$	ICRB0				
FFE6 0488$_H$	RECRB1				
FFE6 048C$_H$	RECRB2				
FFE6 0490$_H$	RECRB3				
FFE6 0494$_H$	RECRB4				
FFE6 0498$_H$	RECRB5				
FFE6 049C$_H$	RECRB6				
FFE6 04A0$_H$	RBURB0				
FFE6 04A4$_H$	RBURB1				
FFE6 04A8$_H$	RBURB2				
FFE6 04AC$_H$	RBURB3				
FFE6 04B0$_H$	RBURB4				
FFE6 04B4$_H$	RBURB5				
FFE6 04B8$_H$	RBURB6				
FFE6 04BC$_H$	OCRB0				
FFE6 04C0$_H$	(Prohibited area)				
FFE6 04C4$_H$	(Prohibited area)	OCRB1	(Prohibited area)	TCNTB1	
FFE6 04C8$_H$	(Prohibited area)	OCRB11	(Prohibited area)	OCRB10	
FFE6 04CC$_H$	(Prohibited area)			OCRB12	
FFE6 04D0$_H$	ICRB1				
FFE6 04D4$_H$	ICRB2				
FFE6 04D8$_H$	ICRB33	ICRB32	ICRB31	ICRB30	
FFE6 04DC$_H$		ICRB36	ICRB35	ICRB34	
FFE6 04E0$_H$ to FFE6 04FC$_H$	(Prohibited area)				
FFE6 0500$_H$	LDB				
FFE6 0504$_H$	RLDB				
FFE6 0508$_H$	TCNTB2				
FFE6 050C$_H$	PIMR2		PIMR1		
FFE6 0510$_H$	TCNTB6				
FFE6 0514$_H$	ICRB6				
FFE6 0518$_H$	(Prohibited area)			RARB6	
FFE6 051C$_H$	TCNTB6M				
FFE6 0520$_H$	OCRB6				
FFE6 0524$_H$	OCRB7				

Timer B Registers (FFE6 0400$_H$ to FFE6 05FF$_H$) (2/2)

Address	b31 +3 b24	b23 +2 b16	b15 +1 b8	b7 +0 b0	Page
FFE6 0528$_H$ to FFE6 053C$_H$	(Prohibited area)				
FFE6 0540$_H$ to FFE6 057C$_H$	(Prohibited area)				
FFE6 0580$_H$	TCNTB3				
FFE6 0584$_H$	TCNTB4				
FFE6 0588$_H$	TCNTB5				
FFE6 058C$_H$	(Prohibited area)	TCCLFCRB	TCCLFSRB	TCCLFRB	
FFE6 0590$_H$	TCCLRB				
FFE6 0594$_H$	OCRB8				
FFE6 0598$_H$	(Prohibited area)				
FFE6 059C$_H$	(Prohibited area)				
FFE6 05A0$_H$		ACRSTRB	ACRCLRB	ACRTRGB	
FFE6 05A4$_H$			ACRVALRB		
FFE6 05A8$_H$ to FFE6 05FC$_H$	(Prohibited area)				

Timer C Registers (FFE6 0600$_H$ to FFE6 0A1F$_H$) (1/5)

Address	b31 +3 b24	b23 +2 b16	b15 +1 b8	b7 +0 b0	Page
FFE6 0600$_H$	(Prohibited area)			TSTRC	Entire C
FFE6 0604$_H$	NCCRC3	NCCRC2	NCCRC1	NCCRC0	
FFE6 0608$_H$	NCCRC7	NCCRC6	NCCRC5	NCCRC4	
FFE6 060C$_H$	(Prohibited area)				
FFE6 0610$_H$		NCMCR2C		NCMCR1C	
FFE6 0614$_H$					
FFE6 0618$_H$					
FFE6 061C$_H$	(Prohibited area)				
FFE6 0620$_H$	TSCRC0		(Prohibited area)		C0
FFE6 0624$_H$	TSRC0		TCRC0		
FFE6 0628$_H$	(Prohibited area)		TIORC0		
FFE6 062C$_H$	(Prohibited area)				
FFE6 0630$_H$	(Prohibited area)				
FFE6 0634$_H$	(Prohibited area)		TIERC0		
FFE6 0638$_H$	TCNTC0				
FFE6 063C$_H$	CUCRC0				
FFE6 0640$_H$	GRC00				
FFE6 0644$_H$	GRC01				
FFE6 0648$_H$	GRC02				
FFE6 064C$_H$	GRC03				
FFE6 0650$_H$	OCRC00				
FFE6 0654$_H$	OCRC01				
FFE6 0658$_H$	OCRC02				
FFE6 065C$_H$	OCRC03				
FFE6 0660$_H$ to FFE6 066C$_H$	(Prohibited area)				
FFE6 0670$_H$	NCRC00		NCNTC00		
FFE6 0674$_H$	NCRC01		NCNTC01		
FFE6 0678$_H$	NCRC02		NCNTC02		
FFE6 067C$_H$	NCRC03		NCNTC03		
FFE6 0680$_H$ to FFE6 069C$_H$	(Prohibited area)				
FFE6 06A0$_H$	TSCRC1		(Prohibited area)		C1
FFE6 06A4$_H$	TSRC1		TCRC1		
FFE6 06A8$_H$	(Prohibited area)		TIORC1		
FFE6 06AC$_H$	(Prohibited area)				
FFE6 06B0$_H$	(Prohibited area)				
FFE6 06B4$_H$	(Prohibited area)		TIERC1		
FFE6 06B8$_H$	TCNTC1				
FFE6 06BC$_H$	CUCRC1				
FFE6 06C0$_H$	GRC10				
FFE6 06C4$_H$	GRC11				

Timer C Registers (FFE6 0600$_H$ to FFE6 0A1F$_H$) (2/5)

Address	b31 +3 b24	b23 +2 b16	b15 +1 b8	b7 +0 b0	Page
FFE6 06C8$_H$	GRC12				
FFE6 06CC$_H$	GRC13				
FFE6 06D0$_H$	OCRC10				
FFE6 06D4$_H$	OCRC11				
FFE6 06D8$_H$	OCRC12				
FFE6 06DC$_H$	OCRC13				
FFE6 06E0$_H$ to FFE6 06EC$_H$	(Prohibited area)				
FFE6 06F0$_H$	NCRC10		NCNTC10		
FFE6 06F4$_H$	NCRC11		NCNTC11		
FFE6 06F8$_H$	NCRC12		NCNTC12		
FFE6 06FC$_H$	NCRC13		NCNTC13		
FFE6 0700$_H$ to FFE6 071C$_H$	(Prohibited area)				
FFE6 0720$_H$	TSCRC2		(Prohibited area)		C2
FFE6 0724$_H$	TSRC2		TCRC2		
FFE6 0728$_H$	(Prohibited area)		TIORC2		
FFE6 072C$_H$	(Prohibited area)				
FFE6 0730$_H$	(Prohibited area)				
FFE6 0734$_H$	(Prohibited area)		TIERC2		
FFE6 0738$_H$	TCNTC2				
FFE6 073C$_H$	CUCRC2				
FFE6 0740$_H$	GRC20				
FFE6 0744$_H$	GRC21				
FFE6 0748$_H$	GRC22				
FFE6 074C$_H$	GRC23				
FFE6 0750$_H$	OCRC20				
FFE6 0754$_H$	OCRC21				
FFE6 0758$_H$	OCRC22				
FFE6 075C$_H$	OCRC23				
FFE6 0760$_H$ to FFE6 076C$_H$	(Prohibited area)				
FFE6 0770$_H$	NCRC20		NCNTC20		
FFE6 0774$_H$	NCRC21		NCNTC21		
FFE6 0778$_H$	NCRC22		NCNTC22		
FFE6 077C$_H$	NCRC23		NCNTC23		
FFE6 0780$_H$ to FFE6 079C$_H$	(Prohibited area)				
FFE6 07A0$_H$	TSCRC3		(Prohibited area)		C3
FFE6 07A4$_H$	TSRC3		TCRC3		

Timer C Registers (FFE6 0600$_H$ to FFE6 0A1F$_H$) (3/5)

Address	b31 +3 b24	b23 +2 b16	b15 +1 b8	b7 +0 b0	Page
FFE6 07A8$_H$	(Prohibited area)		TIORC3		
FFE6 07AC$_H$	(Prohibited area)				
FFE6 07B0$_H$	(Prohibited area)				
FFE6 07B4$_H$	(Prohibited area)		TIERC3		
FFE6 07B8$_H$	TCNTC3				
FFE6 07BC$_H$	CUCRC3				
FFE6 07C0$_H$	GRC30				
FFE6 07C4$_H$	GRC31				
FFE6 07C8$_H$	GRC32				
FFE6 07CC$_H$	GRC33				
FFE6 07D0$_H$	OCRC30				
FFE6 07D4$_H$	OCRC31				
FFE6 07D8$_H$	OCRC32				
FFE6 07DC$_H$	OCRC33				
FFE6 07E0$_H$ to FFE6 07EC$_H$	(Prohibited area)				
FFE6 07F0$_H$	NCRC30		NCNTC30		
FFE6 07F4$_H$	NCRC31		NCNTC31		
FFE6 07F8$_H$	NCRC32		NCNTC32		
FFE6 07FC$_H$	NCRC33		NCNTC33		
FFE6 0800$_H$ to FFE6 081C$_H$	(Prohibited area)				
FFE6 0820$_H$	TSCRC4		(Prohibited area)		C4
FFE6 0824$_H$	TSRC4		TCRC4		
FFE6 0828$_H$	(Prohibited area)		TIORC4		
FFE6 082C$_H$	(Prohibited area)				
FFE6 0830$_H$	(Prohibited area)				
FFE6 0834$_H$	(Prohibited area)		TIERC4		
FFE6 0838$_H$	TCNTC4				
FFE6 083C$_H$	CUCRC4				
FFE6 0840$_H$	GRC40				
FFE6 0844$_H$	GRC41				
FFE6 0848$_H$	GRC42				
FFE6 084C$_H$	GRC43				
FFE6 0850$_H$	OCRC40				
FFE6 0854$_H$	OCRC41				
FFE6 0858$_H$	OCRC42				
FFE6 085C$_H$	OCRC43				
FFE6 0860$_H$ to FFE6 086C$_H$	(Prohibited area)				
FFE6 0870$_H$	NCRC40		NCNTC40		

Timer C Registers (FFE6 0600$_H$ to FFE6 0A1F$_H$) (4/5)

Address	b31 +3 b24	b23 +2 b16	b15 +1 b8	b7 +0 b0	Page
FFE6 0874$_H$	NCRC41		NCNTC41		
FFE6 0878$_H$	NCRC42		NCNTC42		
FFE6 087C$_H$	NCRC43		NCNTC43		
FFE6 0880$_H$ to FFE6 089C$_H$	(Prohibited area)				
FFE6 08A0$_H$	TSCRC5		(Prohibited area)		C5
FFE6 08A4$_H$	TSRC5		TCRC5		
FFE6 08A8$_H$	(Prohibited area)		TIORC5		
FFE6 08AC$_H$	(Prohibited area)				
FFE6 08B0$_H$	(Prohibited area)				
FFE6 08B4$_H$	(Prohibited area)		TIERC5		
FFE6 08B8$_H$	TCNTC5				
FFE6 08BC$_H$	CUCRC5				
FFE6 08C0$_H$	GRC50				
FFE6 08C4$_H$	GRC51				
FFE6 08C8$_H$	GRC52				
FFE6 08CC$_H$	GRC53				
FFE6 08D0$_H$	OCRC50				
FFE6 08D4$_H$	OCRC51				
FFE6 08D8$_H$	OCRC52				
FFE6 08DC$_H$	OCRC53				
FFE6 08E0$_H$ to FFE6 08EC$_H$	(Prohibited area)				
FFE6 08F0$_H$	NCRC50		NCNTC50		
FFE6 08F4$_H$	NCRC51		NCNTC51		
FFE6 08F8$_H$	NCRC52		NCNTC52		
FFE6 08FC$_H$	NCRC53		NCNTC53		
FFE6 0900$_H$ to FFE6 091C$_H$	(Prohibited area)				
FFE6 0920$_H$	TSCRC6		(Prohibited area)		C6
FFE6 0924$_H$	TSRC6		TCRC6		
FFE6 0928$_H$	(Prohibited area)		TIORC6		
FFE6 092C$_H$	(Prohibited area)				
FFE6 0930$_H$	(Prohibited area)				
FFE6 0934$_H$	(Prohibited area)		TIERC6		
FFE6 0938$_H$	TCNTC6				
FFE6 093C$_H$	CUCRC6				
FFE6 0940$_H$	GRC60				
FFE6 0944$_H$	GRC61				
FFE6 0948$_H$	GRC62				
FFE6 094C$_H$	GRC63				

Timer C Registers (FFE6 0600$_H$ to FFE6 0A1F$_H$) (5/5)

Address	b31 +3 b24	b23 +2 b16	b15 +1 b8	b7 +0 b0	Page
FFE6 0950$_H$	OCRC60				
FFE6 0954$_H$	OCRC61				
FFE6 0958$_H$	OCRC62				
FFE6 095C$_H$	OCRC63				
FFE6 0960$_H$ to FFE6 096C$_H$	(Prohibited area)				
FFE6 0970$_H$	NCRC60		NCNTC60		
FFE6 0974$_H$	NCRC61		NCNTC61		
FFE6 0978$_H$	NCRC62		NCNTC62		
FFE6 097C$_H$	NCRC63		NCNTC63		
FFE6 0980$_H$ to FFE6 099C$_H$	(Prohibited area)				
FFE6 09A0$_H$	TSCRC7		(Prohibited area)		C7
FFE6 09A4$_H$	TSRC7		TCRC7		
FFE6 09A8$_H$	(Prohibited area)		TIORC7		
FFE6 09AC$_H$	(Prohibited area)				
FFE6 09B0$_H$	(Prohibited area)				
FFE6 09B4$_H$	(Prohibited area)		TIERC7		
FFE6 09B8$_H$	TCNTC7				
FFE6 09BC$_H$	CUCRC7				
FFE6 09C0$_H$	GRC70				
FFE6 09C4$_H$	GRC71				
FFE6 09C8$_H$	GRC72				
FFE6 09CC$_H$	GRC73				
FFE6 09D0$_H$	OCRC70				
FFE6 09D4$_H$	OCRC71				
FFE6 09D8$_H$	OCRC72				
FFE6 09DC$_H$	OCRC73				
FFE6 09E0$_H$ to FFE6 09EC$_H$	(Prohibited area)				
FFE6 09F0$_H$	NCRC70		NCNTC70		
FFE6 09F4$_H$	NCRC71		NCNTC71		
FFE6 09F8$_H$	NCRC72		NCNTC72		
FFE6 09FC$_H$	NCRC73		NCNTC73		
FFE6 0A00$_H$ to FFE6 0A1C$_H$	(Prohibited area)				

Timer D Registers (FFE6 1000$_H$ to FFE6 1FFC$_H$) (1/12)

Address	b31 +3 b24	b23 +2 b16	b15 +1 b8	b7 +0 b0	Page
FFE6 1000$_H$	(Prohibited area)		TSTRD		Entire D
FFE6 1004$_H$	(Prohibited area)				
to					
FFE6 10F8$_H$					
FFE6 10FC$_H$	(Prohibited area)				
FFE6 1100$_H$	DCRD0		TCRD0		D0
FFE6 1104$_H$	TIOR2D0		TIOR1D0		
FFE6 1108$_H$	DSR2D0	DSR1D0	DSCRD0	DSTRD0	
FFE6 110C$_H$	TSCRD0		TSRD0		
FFE6 1110$_H$	(Prohibited area)	ODRD0	OSELRD0	TOCRD0	
FFE6 1114$_H$	(Prohibited area)			TICTSELD0	
FFE6 1118$_H$	OSBRD0				
FFE6 111C$_H$	(Prohibited area)				
FFE6 1120$_H$	TCNT1D0				
FFE6 1124$_H$	TCNT2D0				
FFE6 1128$_H$	CUCR1D0				
FFE6 112C$_H$	CUCR2D0				
FFE6 1130$_H$	(Prohibited area)			RCR1D0	
FFE6 1134$_H$	(Prohibited area)			RCR2D0	
FFE6 1138$_H$	(Prohibited area)				
FFE6 113C$_H$	(Prohibited area)				
FFE6 1140$_H$	OCR1D00				
FFE6 1144$_H$	(Prohibited area)				
FFE6 1148$_H$	OCR2D00				
FFE6 114C$_H$	(Prohibited area)				
FFE6 1150$_H$	ICR1D00				
FFE6 1154$_H$	ICR2D00				
FFE6 1158$_H$	DCNTD00				
FFE6 115C$_H$	(Prohibited area)				
FFE6 1160$_H$	OCR1D01				
FFE6 1164$_H$	(Prohibited area)				
FFE6 1168$_H$	OCR2D01				
FFE6 116C$_H$	(Prohibited area)				
FFE6 1170$_H$	ICR1D01				
FFE6 1174$_H$	ICR2D01				
FFE6 1178$_H$	DCNTD01				
FFE6 117C$_H$	(Prohibited area)				
FFE6 1180$_H$	OCR1D02				
FFE6 1184$_H$	(Prohibited area)				
FFE6 1188$_H$	OCR2D02				
FFE6 118C$_H$	(Prohibited area)				
FFE6 1190$_H$	ICR1D02				
FFE6 1194$_H$	ICR2D02				
FFE6 1198$_H$	DCNTD02				

Timer D Registers (FFE6 1000$_H$ to FFE6 1FFC$_H$) (2/12)

Address	b31 +3 b24	b23 +2 b16	b15 +1 b8	b7 +0 b0	Page
FFE6 119C$_H$	(Prohibited area)				
FFE6 11A0$_H$	OCR1D03				
FFE6 11A4$_H$	(Prohibited area)				
FFE6 11A8$_H$	OCR2D03				
FFE6 11AC$_H$	(Prohibited area)				
FFE6 11B0$_H$	ICR1D03				
FFE6 11B4$_H$	ICR2D03				
FFE6 11B8$_H$	DCNTD03				
FFE6 11BC$_H$	(Prohibited area)				
FFE6 11C0$_H$	(Prohibited area)				
to					
FFE6 11FC$_H$					
FFE6 1200$_H$	DCRD1		TCRD1		D1
FFE6 1204$_H$	TIOR2D1		TIOR1D1		
FFE6 1208$_H$	DSR2D1	DSR1D1	DSCRD1	DSTRD1	
FFE6 120C$_H$	TSCRD1		TSRD1		
FFE6 1210$_H$	(Prohibited area)	ODRD1	OSELRD1	TOCRD1	
FFE6 1214$_H$	(Prohibited area)			TICTSELD1	
FFE6 1218$_H$	OSBRD1				
FFE6 121C$_H$	(Prohibited area)				
FFE6 1220$_H$	TCNT1D1				
FFE6 1224$_H$	TCNT2D1				
FFE6 1228$_H$	CUCR1D1				
FFE6 122C$_H$	CUCR2D1				
FFE6 1230$_H$	(Prohibited area)			RCR1D1	
FFE6 1234$_H$	(Prohibited area)			RCR2D1	
FFE6 1238$_H$	(Prohibited area)				
FFE6 123C$_H$	(Prohibited area)				
FFE6 1240$_H$	OCR1D10				
FFE6 1244$_H$	(Prohibited area)				
FFE6 1248$_H$	OCR2D10				
FFE6 124C$_H$	(Prohibited area)				
FFE6 1250$_H$	ICR1D10				
FFE6 1254$_H$	ICR2D10				
FFE6 1258$_H$	DCNTD10				
FFE6 125C$_H$	(Prohibited area)				
FFE6 1260$_H$	OCR1D11				
FFE6 1264$_H$	(Prohibited area)				
FFE6 1268$_H$	OCR2D11				
FFE6 126C$_H$	(Prohibited area)				
FFE6 1270$_H$	ICR1D11				
FFE6 1274$_H$	ICR2D11				
FFE6 1278$_H$	DCNTD11				
FFE6 127C$_H$	(Prohibited area)				

Timer D Registers (FFE6 1000$_H$ to FFE6 1FFC$_H$) (3/12)

Address	b31 +3 b24	b23 +2 b16	b15 +1 b8	b7 +0 b0	Page
FFE6 1280$_H$	OCR1D12				
FFE6 1284$_H$	(Prohibited area)				
FFE6 1288$_H$	OCR2D12				
FFE6 128C$_H$	(Prohibited area)				
FFE6 1290$_H$	ICR1D12				
FFE6 1294$_H$	ICR2D12				
FFE6 1298$_H$	DCNTD12				
FFE6 129C$_H$	(Prohibited area)				
FFE6 12A0$_H$	OCR1D13				
FFE6 12A4$_H$	(Prohibited area)				
FFE6 12A8$_H$	OCR2D13				
FFE6 12AC$_H$	(Prohibited area)				
FFE6 12B0$_H$	ICR1D13				
FFE6 12B4$_H$	ICR2D13				
FFE6 12B8$_H$	DCNTD13				
FFE6 12BC$_H$	(Prohibited area)				
FFE6 12C0$_H$	(Prohibited area)				
to					
FFE6 12FC$_H$					
FFE6 1300$_H$	DCRD2		TCRD2		D2
FFE6 1304$_H$	TIOR2D2		TIOR1D2		
FFE6 1308$_H$	DSR2D2	DSR1D2	DSCRD2	DSTRD2	
FFE6 130C$_H$	TSCRD2		TSRD2		
FFE6 1310$_H$	(Prohibited area)	ODRD2	OSELRD2	TOCRD2	
FFE6 1314$_H$	(Prohibited area)			TICTSELD2	
FFE6 1318$_H$	OSBRD2				
FFE6 131C$_H$	(Prohibited area)				
FFE6 1320$_H$	TCNT1D2				
FFE6 1324$_H$	TCNT2D2				
FFE6 1328$_H$	CUCR1D2				
FFE6 132C$_H$	CUCR2D2				
FFE6 1330$_H$	(Prohibited area)			RCR1D2	
FFE6 1334$_H$	(Prohibited area)			RCR2D2	
FFE6 1338$_H$	(Prohibited area)				
FFE6 133C$_H$	(Prohibited area)				
FFE6 1340$_H$	OCR1D20				
FFE6 1344$_H$	(Prohibited area)				
FFE6 1348$_H$	OCR2D20				
FFE6 134C$_H$	(Prohibited area)				
FFE6 1350$_H$	ICR1D20				
FFE6 1354$_H$	ICR2D20				
FFE6 1358$_H$	DCNTD20				
FFE6 135C$_H$	(Prohibited area)				
FFE6 1360$_H$	OCR1D21				

Timer D Registers (FFE6 1000$_H$ to FFE6 1FFC$_H$) (4/12)

Address	b31 +3 b24	b23 +2 b16	b15 +1 b8	b7 +0 b0	Page
FFE6 1364$_H$	(Prohibited area)				
FFE6 1368$_H$	OCR2D21				
FFE6 136C$_H$	(Prohibited area)				
FFE6 1370$_H$	ICR1D21				
FFE6 1374$_H$	ICR2D21				
FFE6 1378$_H$	DCNTD21				
FFE6 137C$_H$	(Prohibited area)				
FFE6 1380$_H$	OCR1D22				
FFE6 1384$_H$	(Prohibited area)				
FFE6 1388$_H$	OCR2D22				
FFE6 138C$_H$	(Prohibited area)				
FFE6 1390$_H$	ICR1D22				
FFE6 1394$_H$	ICR2D22				
FFE6 1398$_H$	DCNTD22				
FFE6 139C$_H$	(Prohibited area)				
FFE6 13A0$_H$	OCR1D23				
FFE6 13A4$_H$	(Prohibited area)				
FFE6 13A8$_H$	OCR2D23				
FFE6 13AC$_H$	(Prohibited area)				
FFE6 13B0$_H$	ICR1D23				
FFE6 13B4$_H$	ICR2D23				
FFE6 13B8$_H$	DCNTD23				
FFE6 13BC$_H$	(Prohibited area)				
FFE6 13C0$_H$ to FFE6 13FC$_H$	(Prohibited area)				
FFE6 1400$_H$	DCRD3		TCRD3		D3
FFE6 1404$_H$	TIOR2D3		TIOR1D3		
FFE6 1408$_H$	DSR2D3	DSR1D3	DSCRD3	DSTRD3	
FFE6 140C$_H$	TSCRD3		TSRD3		
FFE6 1410$_H$	(Prohibited area)	ODRD3	OSELRD3	TOCRD3	
FFE6 1414$_H$	(Prohibited area)			TICTSELD3	
FFE6 1418$_H$	OSBRD3				
FFE6 141C$_H$	(Prohibited area)				
FFE6 1420$_H$	TCNT1D3				
FFE6 1424$_H$	TCNT2D3				
FFE6 1428$_H$	CUCR1D3				
FFE6 142C$_H$	CUCR2D3				
FFE6 1430$_H$	(Prohibited area)			RCR1D3	
FFE6 1434$_H$	(Prohibited area)			RCR2D3	
FFE6 1438$_H$	(Prohibited area)				
FFE6 143C$_H$	(Prohibited area)				
FFE6 1440$_H$	OCR1D30				
FFE6 1444$_H$	(Prohibited area)				

Timer D Registers (FFE6 1000$_H$ to FFE6 1FFC$_H$) (5/12)

Address	b31 +3 b24	b23 +2 b16	b15 +1 b8	b7 +0 b0	Page
FFE6 1448$_H$	OCR2D30				
FFE6 144C$_H$	(Prohibited area)				
FFE6 1450$_H$	ICR1D30				
FFE6 1454$_H$	ICR2D30				
FFE6 1458$_H$	DCNTD30				
FFE6 145C$_H$	(Prohibited area)				
FFE6 1460$_H$	OCR1D31				
FFE6 1464$_H$	(Prohibited area)				
FFE6 1468$_H$	OCR2D31				
FFE6 146C$_H$	(Prohibited area)				
FFE6 1470$_H$	ICR1D31				
FFE6 1474$_H$	ICR2D31				
FFE6 1478$_H$	DCNTD31				
FFE6 147C$_H$	(Prohibited area)				
FFE6 1480$_H$	OCR1D32				
FFE6 1484$_H$	(Prohibited area)				
FFE6 1488$_H$	OCR2D32				
FFE6 148C$_H$	(Prohibited area)				
FFE6 1490$_H$	ICR1D32				
FFE6 1494$_H$	ICR2D32				
FFE6 1498$_H$	DCNTD32				
FFE6 149C$_H$	(Prohibited area)				
FFE6 14A0$_H$	OCR1D33				
FFE6 14A4$_H$	(Prohibited area)				
FFE6 14A8$_H$	OCR2D33				
FFE6 14AC$_H$	(Prohibited area)				
FFE6 14B0$_H$	ICR1D33				
FFE6 14B4$_H$	ICR2D33				
FFE6 14B8$_H$	DCNTD33				
FFE6 14BC$_H$	(Prohibited area)				
FFE6 14C0$_H$	(Prohibited area)				
to					
FFE6 14FC$_H$					
FFE6 1500$_H$	DCRD4		TCRD4		D4
FFE6 1504$_H$	TIOR2D4		TIOR1D4		
FFE6 1508$_H$	DSR2D4	DSR1D4	DSCRD4	DSTRD4	
FFE6 150C$_H$	TSCRD4		TSRD4		
FFE6 1510$_H$	(Prohibited area)	ODRD4	OSELRD4	TOCRD4	
FFE6 1514$_H$	(Prohibited area)			TICTSELD4	
FFE6 1518$_H$	OSBRD4				
FFE6 151C$_H$	(Prohibited area)				
FFE6 1520$_H$	TCNT1D4				
FFE6 1524$_H$	TCNT2D4				
FFE6 1528$_H$	CUCR1D4				

Timer D Registers (FFE6 1000$_H$ to FFE6 1FFC$_H$) (6/12)

Address	b31 +3 b24	b23 +2 b16	b15 +1 b8	b7 +0 b0	Page
FFE6 152C$_H$	CUCR2D4				
FFE6 1530$_H$	(Prohibited area)			RCR1D4	
FFE6 1534$_H$	(Prohibited area)			RCR2D4	
FFE6 1538$_H$	(Prohibited area)				
FFE6 153C$_H$	(Prohibited area)				
FFE6 1540$_H$	OCR1D40				
FFE6 1544$_H$	(Prohibited area)				
FFE6 1548$_H$	OCR2D40				
FFE6 154C$_H$	(Prohibited area)				
FFE6 1550$_H$	ICR1D40				
FFE6 1554$_H$	ICR2D40				
FFE6 1558$_H$	DCNTD40				
FFE6 155C$_H$	(Prohibited area)				
FFE6 1560$_H$	OCR1D41				
FFE6 1564$_H$	(Prohibited area)				
FFE6 1568$_H$	OCR2D41				
FFE6 156C$_H$	(Prohibited area)				
FFE6 1570$_H$	ICR1D41				
FFE6 1574$_H$	ICR2D41				
FFE6 1578$_H$	DCNTD41				
FFE6 157C$_H$	(Prohibited area)				
FFE6 1580$_H$	OCR1D42				
FFE6 1584$_H$	(Prohibited area)				
FFE6 1588$_H$	OCR2D42				
FFE6 158C$_H$	(Prohibited area)				
FFE6 1590$_H$	ICR1D42				
FFE6 1594$_H$	ICR2D42				
FFE6 1598$_H$	DCNTD42				
FFE6 159C$_H$	(Prohibited area)				
FFE6 15A0$_H$	OCR1D43				
FFE6 15A4$_H$	(Prohibited area)				
FFE6 15A8$_H$	OCR2D43				
FFE6 15AC$_H$	(Prohibited area)				
FFE6 15B0$_H$	ICR1D43				
FFE6 15B4$_H$	ICR2D43				
FFE6 15B8$_H$	DCNTD43				
FFE6 15BC$_H$	(Prohibited area)				
FFE6 15C0$_H$	(Prohibited area)				
to					
FFE6 15FC$_H$					
FFE6 1600$_H$	DCRD5		TCRD5		D5
FFE6 1604$_H$	TIOR2D5		TIOR1D5		
FFE6 1608$_H$	DSR2D5	DSR1D5	DSCRD5	DSTRD5	
FFE6 160C$_H$	TSCRD5		TSRD5		

Timer D Registers (FFE6 1000$_H$ to FFE6 1FFC$_H$) (7/12)

Address	b31 +3 b24	b23 +2 b16	b15 +1 b8	b7 +0 b0	Page
FFE6 1610$_H$	(Prohibited area)	ODRD5	OSELRD5	TOCRD5	
FFE6 1614$_H$	(Prohibited area)			TICTSELD5	
FFE6 1618$_H$	OSBRD5				
FFE6 161C$_H$	(Prohibited area)				
FFE6 1620$_H$	TCNT1D5				
FFE6 1624$_H$	TCNT2D5				
FFE6 1628$_H$	CUCR1D5				
FFE6 162C$_H$	CUCR2D5				
FFE6 1630$_H$	(Prohibited area)			RCR1D5	
FFE6 1634$_H$	(Prohibited area)			RCR2D5	
FFE6 1638$_H$	(Prohibited area)				
FFE6 163C$_H$	(Prohibited area)				
FFE6 1640$_H$	OCR1D50				
FFE6 1644$_H$	(Prohibited area)				
FFE6 1648$_H$	OCR2D50				
FFE6 164C$_H$	(Prohibited area)				
FFE6 1650$_H$	ICR1D50				
FFE6 1654$_H$	ICR2D50				
FFE6 1658$_H$	DCNTD50				
FFE6 165C$_H$	(Prohibited area)				
FFE6 1660$_H$	OCR1D51				
FFE6 1664$_H$	(Prohibited area)				
FFE6 1668$_H$	OCR2D51				
FFE6 166C$_H$	(Prohibited area)				
FFE6 1670$_H$	ICR1D51				
FFE6 1674$_H$	ICR2D51				
FFE6 1678$_H$	DCNTD51				
FFE6 167C$_H$	(Prohibited area)				
FFE6 1680$_H$	OCR1D52				
FFE6 1684$_H$	(Prohibited area)				
FFE6 1688$_H$	OCR2D52				
FFE6 168C$_H$	(Prohibited area)				
FFE6 1690$_H$	ICR1D52				
FFE6 1694$_H$	ICR2D52				
FFE6 1698$_H$	DCNTD52				
FFE6 169C$_H$	(Prohibited area)				
FFE6 16A0$_H$	OCR1D53				
FFE6 16A4$_H$	(Prohibited area)				
FFE6 16A8$_H$	OCR2D53				
FFE6 16AC$_H$	(Prohibited area)				
FFE6 16B0$_H$	ICR1D53				
FFE6 16B4$_H$	ICR2D53				
FFE6 16B8$_H$	DCNTD53				
FFE6 16BC$_H$	(Prohibited area)				

Timer D Registers (FFE6 1000$_H$ to FFE6 1FFC$_H$) (8/12)

Address	b31 +3 b24	b23 +2 b16	b15 +1 b8	b7 +0 b0	Page
FFE6 16C0$_H$	(Prohibited area)				
to					
FFE6 16FC$_H$					
FFE6 1700$_H$	DCRD6		TCRD6		D6
FFE6 1704$_H$	TIOR2D6		TIOR1D6		
FFE6 1708$_H$	DSR2D6	DSR1D6	DSCRD6	DSTRD6	
FFE6 170C$_H$	TSCRD6		TSRD6		
FFE6 1710$_H$	(Prohibited area)	ODRD6	OSELRD6	TOCRD6	
FFE6 1714$_H$	(Prohibited area)			TICTSELD6	
FFE6 1718$_H$	OSBRD6				
FFE6 171C$_H$	(Prohibited area)				
FFE6 1720$_H$	TCNT1D6				
FFE6 1724$_H$	TCNT2D6				
FFE6 1728$_H$	CUCR1D6				
FFE6 172C$_H$	CUCR2D6				
FFE6 1730$_H$	(Prohibited area)			RCR1D6	
FFE6 1734$_H$	(Prohibited area)			RCR2D6	
FFE6 1738$_H$	(Prohibited area)				
FFE6 173C$_H$	(Prohibited area)				
FFE6 1740$_H$	OCR1D60				
FFE6 1744$_H$	(Prohibited area)				
FFE6 1748$_H$	OCR2D60				
FFE6 174C$_H$	(Prohibited area)				
FFE6 1750$_H$	ICR1D60				
FFE6 1754$_H$	ICR2D60				
FFE6 1758$_H$	DCNTD60				
FFE6 175C$_H$	(Prohibited area)				
FFE6 1760$_H$	OCR1D61				
FFE6 1764$_H$	(Prohibited area)				
FFE6 1768$_H$	OCR2D61				
FFE6 176C$_H$	(Prohibited area)				
FFE6 1770$_H$	ICR1D61				
FFE6 1774$_H$	ICR2D61				
FFE6 1778$_H$	DCNTD61				
FFE6 177C$_H$	(Prohibited area)				
FFE6 1780$_H$	OCR1D62				
FFE6 1784$_H$	(Prohibited area)				
FFE6 1788$_H$	OCR2D62				
FFE6 178C$_H$	(Prohibited area)				
FFE6 1790$_H$	ICR1D62				
FFE6 1794$_H$	ICR2D62				
FFE6 1798$_H$	DCNTD62				
FFE6 179C$_H$	(Prohibited area)				
FFE6 17A0$_H$	OCR1D63				

Timer D Registers (FFE6 1000$_H$ to FFE6 1FFC$_H$) (9/12)

Address	b31 +3 b24	b23 +2 b16	b15 +1 b8	b7 +0 b0	Page
FFE6 17A4$_H$	(Prohibited area)				
FFE6 17A8$_H$	OCR2D63				
FFE6 17AC$_H$	(Prohibited area)				
FFE6 17B0$_H$	ICR1D63				
FFE6 17B4$_H$	ICR2D63				
FFE6 17B8$_H$	DCNTD63				
FFE6 17BC$_H$	(Prohibited area)				
FFE6 17C0$_H$	(Prohibited area)				
to					
FFE6 17FC$_H$					
FFE6 1800$_H$	DCRD7		TCRD7		D7
FFE6 1804$_H$	TIOR2D7		TIOR1D7		
FFE6 1808$_H$	DSR2D7	DSR1D7	DSCRD7	DSTRD7	
FFE6 180C$_H$	TSCRD7		TSRD7		
FFE6 1810$_H$	(Prohibited area)	ODRD7	OSELRD7	TOCRD7	
FFE6 1814$_H$	(Prohibited area)			TICTSELD7	
FFE6 1818$_H$	OSBRD7				
FFE6 181C$_H$	(Prohibited area)				
FFE6 1820$_H$	TCNT1D7				
FFE6 1824$_H$	TCNT2D7				
FFE6 1828$_H$	CUCR1D7				
FFE6 182C$_H$	CUCR2D7				
FFE6 1830$_H$	(Prohibited area)			RCR1D7	
FFE6 1834$_H$	(Prohibited area)			RCR2D7	
FFE6 1838$_H$	(Prohibited area)				
FFE6 183C$_H$	(Prohibited area)				
FFE6 1840$_H$	OCR1D70				
FFE6 1844$_H$	(Prohibited area)				
FFE6 1848$_H$	OCR2D70				
FFE6 184C$_H$	(Prohibited area)				
FFE6 1850$_H$	ICR1D70				
FFE6 1854$_H$	ICR2D70				
FFE6 1858$_H$	DCNTD70				
FFE6 185C$_H$	(Prohibited area)				
FFE6 1860$_H$	OCR1D71				
FFE6 1864$_H$	(Prohibited area)				
FFE6 1868$_H$	OCR2D71				
FFE6 186C$_H$	(Prohibited area)				
FFE6 1870$_H$	ICR1D71				
FFE6 1874$_H$	ICR2D71				
FFE6 1878$_H$	DCNTD71				
FFE6 187C$_H$	(Prohibited area)				
FFE6 1880$_H$	OCR1D72				
FFE6 1884$_H$	(Prohibited area)				

Timer D Registers (FFE6 1000$_H$ to FFE6 1FFC$_H$) (10/12)

Address	b31 +3 b24	b23 +2 b16	b15 +1 b8	b7 +0 b0	Page
FFE6 1888$_H$	OCR2D72				
FFE6 188C$_H$	(Prohibited area)				
FFE6 1890$_H$	ICR1D72				
FFE6 1894$_H$	ICR2D72				
FFE6 1898$_H$	DCNTD72				
FFE6 189C$_H$	(Prohibited area)				
FFE6 18A0$_H$	OCR1D73				
FFE6 18A4$_H$	(Prohibited area)				
FFE6 18A8$_H$	OCR2D73				
FFE6 18AC$_H$	(Prohibited area)				
FFE6 18B0$_H$	ICR1D73				
FFE6 18B4$_H$	ICR2D73				
FFE6 18B8$_H$	DCNTD73				
FFE6 18BC$_H$	(Prohibited area)				
FFE6 18C0$_H$ to FFE6 18FC$_H$	(Prohibited area)				
FFE6 1900$_H$	DCRD8		TCRD8		D8
FFE6 1904$_H$	TIOR2D8		TIOR1D8		
FFE6 1908$_H$	DSR2D8	DSR1D8	DSCRD8	DSTRD8	
FFE6 190C$_H$	TSCRD8		TSRD8		
FFE6 1910$_H$	(Prohibited area)	ODRD8	OSELRD8	TOCRD8	
FFE6 1914$_H$	(Prohibited area)			TICTSELD8	
FFE6 1918$_H$	OSBRD8				
FFE6 191C$_H$	(Prohibited area)				
FFE6 1920$_H$	TCNT1D8				
FFE6 1924$_H$	TCNT2D8				
FFE6 1928$_H$	CUCR1D8				
FFE6 192C$_H$	CUCR2D8				
FFE6 1930$_H$	(Prohibited area)			RCR1D8	
FFE6 1934$_H$	(Prohibited area)			RCR2D8	
FFE6 1938$_H$	(Prohibited area)				
FFE6 193C$_H$	(Prohibited area)				
FFE6 1940$_H$	OCR1D80				
FFE6 1944$_H$	(Prohibited area)				
FFE6 1948$_H$	OCR2D80				
FFE6 194C$_H$	(Prohibited area)				
FFE6 1950$_H$	ICR1D80				
FFE6 1954$_H$	ICR2D80				
FFE6 1958$_H$	DCNTD80				
FFE6 195C$_H$	(Prohibited area)				
FFE6 1960$_H$	OCR1D81				
FFE6 1964$_H$	(Prohibited area)				
FFE6 1968$_H$	OCR2D81				

Timer D Registers (FFE6 1000$_H$ to FFE6 1FFC$_H$) (11/12)

Address	b31 +3 b24	b23 +2 b16	b15 +1 b8	b7 +0 b0	Page
FFE6 196C$_H$	(Prohibited area)				
FFE6 1970$_H$	ICR1D81				
FFE6 1974$_H$	ICR2D81				
FFE6 1978$_H$	DCNTD81				
FFE6 197C$_H$	(Prohibited area)				
FFE6 1980$_H$	OCR1D82				
FFE6 1984$_H$	(Prohibited area)				
FFE6 1988$_H$	OCR2D82				
FFE6 198C$_H$	(Prohibited area)				
FFE6 1990$_H$	ICR1D82				
FFE6 1994$_H$	ICR2D82				
FFE6 1998$_H$	DCNTD82				
FFE6 199C$_H$	(Prohibited area)				
FFE6 19A0$_H$	OCR1D83				
FFE6 19A4$_H$	(Prohibited area)				
FFE6 19A8$_H$	OCR2D83				
FFE6 19AC$_H$	(Prohibited area)				
FFE6 19B0$_H$	ICR1D83				
FFE6 19B4$_H$	ICR2D83				
FFE6 19B8$_H$	DCNTD83				
FFE6 19BC$_H$	(Prohibited area)				
FFE6 19C0$_H$ to FFE6 19FC$_H$	(Prohibited area)				
FFE6 1A00$_H$	DCRD9		TCRD9		D9
FFE6 1A04$_H$	TIOR2D9		TIOR1D9		
FFE6 1A08$_H$	DSR2D9	DSR1D9	DSCRD9	DSTRD9	
FFE6 1A0C$_H$	TSCRD9		TSRD9		
FFE6 1A10$_H$	(Prohibited area)	ODRD9	OSELRD9	TOCRD9	
FFE6 1A14$_H$	(Prohibited area)			TICTSELD9	
FFE6 1A18$_H$	OSBRD9				
FFE6 1A1C$_H$	(Prohibited area)				
FFE6 1A20$_H$	TCNT1D9				
FFE6 1A24$_H$	TCNT2D9				
FFE6 1A28$_H$	CUCR1D9				
FFE6 1A2C$_H$	CUCR2D9				
FFE6 1A30$_H$	(Prohibited area)			RCR1D9	
FFE6 1A34$_H$	(Prohibited area)			RCR2D9	
FFE6 1A38$_H$	(Prohibited area)				
FFE6 1A3C$_H$	(Prohibited area)				
FFE6 1A40$_H$	OCR1D90				
FFE6 1A44$_H$	(Prohibited area)				
FFE6 1A48$_H$	OCR2D90				
FFE6 1A4C$_H$	(Prohibited area)				

Timer D Registers (FFE6 1000_H to FFE6 $1FFC_H$) (12/12)

Address	b31 +3 b24	b23 +2 b16	b15 +1 b8	b7 +0 b0	Page
FFE6 $1A50_H$	ICR1D90				
FFE6 $1A54_H$	ICR2D90				
FFE6 $1A58_H$	DCNTD90				
FFE6 $1A5C_H$	(Prohibited area)				
FFE6 $1A60_H$	OCR1D91				
FFE6 $1A64_H$	(Prohibited area)				
FFE6 $1A68_H$	OCR2D91				
FFE6 $1A6C_H$	(Prohibited area)				
FFE6 $1A70_H$	ICR1D91				
FFE6 $1A74_H$	ICR2D91				
FFE6 $1A78_H$	DCNTD91				
FFE6 $1A7C_H$	(Prohibited area)				
FFE6 $1A80_H$	OCR1D92				
FFE6 $1A84_H$	(Prohibited area)				
FFE6 $1A88_H$	OCR2D92				
FFE6 $1A8C_H$	(Prohibited area)				
FFE6 $1A90_H$	ICR1D92				
FFE6 $1A94_H$	ICR2D92				
FFE6 $1A98_H$	DCNTD92				
FFE6 $1A9C_H$	(Prohibited area)				
FFE6 $1AA0_H$	OCR1D93				
FFE6 $1A4_H$	(Prohibited area)				
FFE6 $1AA8_H$	OCR2D93				
FFE6 $1AAC_H$	(Prohibited area)				
FFE6 $1AB0_H$	ICR1D93				
FFE6 $1AB4_H$	ICR2D93				
FFE6 $1AB8_H$	DCNTD93				
FFE6 $1ABC_H$	(Prohibited area)				
FFE6 $1AC0_H$	(Prohibited area)				
to					
FFE6 $1FFC_H$					

Timer E Registers (FFE6 2000$_H$ to FFE6 2FFF$_H$) (1/8)

Address	b31 +3 b24	b23 +2 b16	b15 +1 b8	b7 +0 b0	Page
FFE6 2000$_H$	(Prohibited area)			TSTRE	Entire E
FFE6 2004$_H$	(Prohibited area)				
FFE6 2008$_H$	(Prohibited area)				
to					
FFE6 20FC$_H$					
FFE6 2100$_H$	(Prohibited area)			SSTRE0	E0
FFE6 2104$_H$	(Prohibited area)			PSCRE0	
FFE6 2108$_H$	(Prohibited area)	RLDCRE0	(Prohibited area)	TCRE0	
FFE6 210C$_H$	(Prohibited area)	SOLVLE0	POECRE0		
FFE6 2110$_H$	TSCRE0		TSRE0		
FFE6 2114$_H$	TIERE0		(Prohibited area)	TOCRE0	
FFE6 2118$_H$	PSCCRE03	PSCCRE02	PSCCRE01	PSCCRE00	
FFE6 211C$_H$	(Prohibited area)				
FFE6 2120$_H$	(Prohibited area)				
FFE6 2124$_H$	TCNTE00				
FFE6 2128$_H$	CYLRE00				
FFE6 212C$_H$	DTRE00				
FFE6 2130$_H$	CRLDE00				
FFE6 2134$_H$	DRLDE00				
FFE6 2138$_H$	(Prohibited area)				
FFE6 213C$_H$	(Prohibited area)				
FFE6 2140$_H$	(Prohibited area)				
FFE6 2144$_H$	TCNTE01				
FFE6 2148$_H$	CYLRE01				
FFE6 214C$_H$	DTRE01				
FFE6 2150$_H$	CRLDE01				
FFE6 2154$_H$	DRLDE01				
FFE6 2158$_H$	(Prohibited area)				
FFE6 215C$_H$	(Prohibited area)				
FFE6 2160$_H$	(Prohibited area)				
FFE6 2164$_H$	TCNTE02				
FFE6 2168$_H$	CYLRE02				
FFE6 216C$_H$	DTRE02				
FFE6 2170$_H$	CRLDE02				
FFE6 2174$_H$	DRLDE02				
FFE6 2178$_H$	(Prohibited area)				
FFE6 217C$_H$	(Prohibited area)				
FFE6 2180$_H$	(Prohibited area)				
FFE6 2184$_H$	TCNTE03				
FFE6 2188$_H$	CYLRE03				
FFE6 218C$_H$	DTRE03				
FFE6 2190$_H$	CRLDE03				
FFE6 2194$_H$	DRLDE03				
FFE6 2198$_H$	(Prohibited area)				

Timer E Registers (FFE6 2000$_H$ to FFE6 2FFF$_H$) (2/8)

Address	b31 +3 b24	b23 +2 b16	b15 +1 b8	b7 +0 b0	Page
FFE6 219C$_H$	(Prohibited area)				
FFE6 21A0$_H$	(Prohibited area)				
to					
FFE6 21FC$_H$					
FFE6 2200$_H$	(Prohibited area)			SSTRE1	E1
FFE6 2204$_H$	(Prohibited area)			PSCRE1	
FFE6 2208$_H$	(Prohibited area)	RLDCRE1	(Prohibited area)	TCRE1	
FFE6 220C$_H$	(Prohibited area)	SOLVLE1	POECRE1		
FFE6 2210$_H$	TSCRE1		TSRE1		
FFE6 2214$_H$	TIERE1		(Prohibited area)	TOCRE1	
FFE6 2218$_H$	PSCCRE13	PSCCRE12	PSCCRE11	PSCCRE10	
FFE6 221C$_H$	(Prohibited area)				
FFE6 2220$_H$	(Prohibited area)				
FFE6 2224$_H$	TCNTE10				
FFE6 2228$_H$	CYLRE10				
FFE6 222C$_H$	DTRE10				
FFE6 2230$_H$	CRLDE10				
FFE6 2234$_H$	DRLDE10				
FFE6 2238$_H$	(Prohibited area)				
FFE6 223C$_H$	(Prohibited area)				
FFE6 2240$_H$	(Prohibited area)				
FFE6 2244$_H$	TCNTE11				
FFE6 2248$_H$	CYLRE11				
FFE6 224C$_H$	DTRE11				
FFE6 2250$_H$	CRLDE11				
FFE6 2254$_H$	DRLDE11				
FFE6 2258$_H$	(Prohibited area)				
FFE6 225C$_H$	(Prohibited area)				
FFE6 2260$_H$	(Prohibited area)				
FFE6 2264$_H$	TCNTE12				
FFE6 2268$_H$	CYLRE12				
FFE6 226C$_H$	DTRE12				
FFE6 2270$_H$	CRLDE12				
FFE6 2274$_H$	DRLDE12				
FFE6 2278$_H$	(Prohibited area)				
FFE6 227C$_H$	(Prohibited area)				
FFE6 2280$_H$	(Prohibited area)				
FFE6 2284$_H$	TCNTE13				
FFE6 2288$_H$	CYLRE13				
FFE6 228C$_H$	DTRE13				
FFE6 2290$_H$	CRLDE13				
FFE6 2294$_H$	DRLDE13				
FFE6 2298$_H$	(Prohibited area)				
FFE6 229C$_H$	(Prohibited area)				

Timer E Registers (FFE6 2000$_H$ to FFE6 2FFF$_H$) (3/8)

Address	b31 +3 b24	b23 +2 b16	b15 +1 b8	b7 +0 b0	Page
FFE6 22A0$_H$	(Prohibited area)				
to					
FFE6 22FC$_H$					
FFE6 2300$_H$	(Prohibited area)			SSTRE2	E2
FFE6 2304$_H$	(Prohibited area)			PSCRE2	
FFE6 2308$_H$	(Prohibited area)	RLDCRE2	(Prohibited area)	TCRE2	
FFE6 230C$_H$	(Prohibited area)	SOLVLE2	POECRE2		
FFE6 2310$_H$	TSCRE2		TSRE2		
FFE6 2314$_H$	TIERE2		(Prohibited area)	TOCRE2	
FFE6 2318$_H$	PSCCRE23	PSCCRE22	PSCCRE21	PSCCRE20	
FFE6 231C$_H$	(Prohibited area)				
FFE6 2320$_H$	(Prohibited area)				
FFE6 2324$_H$	TCNTE20				
FFE6 2328$_H$	CYLRE20				
FFE6 232C$_H$	DTRE20				
FFE6 2330$_H$	CRLDE20				
FFE6 2334$_H$	DRLDE20				
FFE6 2338$_H$	(Prohibited area)				
FFE6 233C$_H$	(Prohibited area)				
FFE6 2340$_H$	(Prohibited area)				
FFE6 2344$_H$	TCNTE21				
FFE6 2348$_H$	CYLRE21				
FFE6 234C$_H$	DTRE21				
FFE6 2350$_H$	CRLDE21				
FFE6 2354$_H$	DRLDE21				
FFE6 2358$_H$	(Prohibited area)				
FFE6 235C$_H$	(Prohibited area)				
FFE6 2360$_H$	(Prohibited area)				
FFE6 2364$_H$	TCNTE22				
FFE6 2368$_H$	CYLRE22				
FFE6 236C$_H$	DTRE22				
FFE6 2370$_H$	CRLDE22				
FFE6 2374$_H$	DRLDE22				
FFE6 2378$_H$	(Prohibited area)				
FFE6 237C$_H$	(Prohibited area)				
FFE6 2380$_H$	(Prohibited area)				
FFE6 2384$_H$	TCNTE23				
FFE6 2388$_H$	CYLRE23				
FFE6 238C$_H$	DTRE23				
FFE6 2390$_H$	CRLDE23				
FFE6 2394$_H$	DRLDE23				
FFE6 2398$_H$	(Prohibited area)				
FFE6 239C$_H$	(Prohibited area)				

Timer E Registers (FFE6 2000$_H$ to FFE6 2FFF$_H$) (4/8)

Address	b31 +3 b24	b23 +2 b16	b15 +1 b8	b7 +0 b0	Page
FFE6 23A0$_H$	(Prohibited area)				
to					
FFE6 23FC$_H$					
FFE6 2400$_H$	(Prohibited area)			SSTRE3	E3
FFE6 2404$_H$	(Prohibited area)			PSCRE3	
FFE6 2408$_H$	(Prohibited area)	RLDCRE3	(Prohibited area)	TCRE3	
FFE6 240C$_H$	(Prohibited area)	SOLVLE3	POECRE3		
FFE6 2410$_H$	TSCRE3		TSRE3		
FFE6 2414$_H$	TIERE3		(Prohibited area)	TOCRE3	
FFE6 2418$_H$	PSCCRE33	PSCCRE32	PSCCRE31	PSCCRE30	
FFE6 241C$_H$	(Prohibited area)				
FFE6 2420$_H$	(Prohibited area)				
FFE6 2424$_H$	TCNTE30				
FFE6 2428$_H$	CYLRE30				
FFE6 242C$_H$	DTRE30				
FFE6 2430$_H$	CRLDE30				
FFE6 2434$_H$	DRLDE30				
FFE6 2438$_H$	(Prohibited area)				
FFE6 243C$_H$	(Prohibited area)				
FFE6 2440$_H$	(Prohibited area)				
FFE6 2444$_H$	TCNTE31				
FFE6 2448$_H$	CYLRE31				
FFE6 244C$_H$	DTRE31				
FFE6 2450$_H$	CRLDE31				
FFE6 2454$_H$	DRLDE31				
FFE6 2458$_H$	(Prohibited area)				
FFE6 245C$_H$	(Prohibited area)				
FFE6 2460$_H$	(Prohibited area)				
FFE6 2464$_H$	TCNTE32				
FFE6 2468$_H$	CYLRE32				
FFE6 246C$_H$	DTRE32				
FFE6 2470$_H$	CRLDE32				
FFE6 2474$_H$	DRLDE32				
FFE6 2478$_H$	(Prohibited area)				
FFE6 247C$_H$	(Prohibited area)				
FFE6 2480$_H$	(Prohibited area)				
FFE6 2484$_H$	TCNTE33				
FFE6 2488$_H$	CYLRE33				
FFE6 248C$_H$	DTRE33				
FFE6 2490$_H$	CRLDE33				
FFE6 2494$_H$	DRLDE33				
FFE6 2498$_H$	(Prohibited area)				
FFE6 249C$_H$	(Prohibited area)				

Timer E Registers (FFE6 2000$_H$ to FFE6 2FFF$_H$) (5/8)

Address	b31 +3 b24	b23 +2 b16	b15 +1 b8	b7 +0 b0	Page
FFE6 24A0$_H$	(Prohibited area)				
to					
FFE6 24FC$_H$					
FFE6 2500$_H$	(Prohibited area)			SSTRE4	E4
FFE6 2504$_H$	(Prohibited area)			PSCRE4	
FFE6 2508$_H$	(Prohibited area)	RLDCRE4	(Prohibited area)	TCRE4	
FFE6 250C$_H$	(Prohibited area)	SOLVLE4	POECRE4		
FFE6 2510$_H$	TSCRE4		TSRE4		
FFE6 2514$_H$	TIERE4		(Prohibited area)	TOCRE4	
FFE6 2518$_H$	PSCCRE43	PSCCRE42	PSCCRE41	PSCCRE40	
FFE6 251C$_H$	(Prohibited area)				
FFE6 2520$_H$	(Prohibited area)				
FFE6 2524$_H$	TCNTE40				
FFE6 2528$_H$	CYLRE40				
FFE6 252C$_H$	DTRE40				
FFE6 2530$_H$	CRLDE40				
FFE6 2534$_H$	DRLDE40				
FFE6 2538$_H$	(Prohibited area)				
FFE6 253C$_H$	(Prohibited area)				
FFE6 2540$_H$	(Prohibited area)				
FFE6 2544$_H$	TCNTE41				
FFE6 2548$_H$	CYLRE41				
FFE6 254C$_H$	DTRE41				
FFE6 2550$_H$	CRLDE41				
FFE6 2554$_H$	DRLDE41				
FFE6 2558$_H$	(Prohibited area)				
FFE6 255C$_H$	(Prohibited area)				
FFE6 2560$_H$	(Prohibited area)				
FFE6 2564$_H$	TCNTE42				
FFE6 2568$_H$	CYLRE42				
FFE6 256C$_H$	DTRE42				
FFE6 2570$_H$	CRLDE42				
FFE6 2574$_H$	DRLDE42				
FFE6 2578$_H$	(Prohibited area)				
FFE6 257C$_H$	(Prohibited area)				
FFE6 2580$_H$	(Prohibited area)				
FFE6 2584$_H$	TCNTE43				
FFE6 2588$_H$	CYLRE43				
FFE6 258C$_H$	DTRE43				
FFE6 2590$_H$	CRLDE43				
FFE6 2594$_H$	DRLDE43				
FFE6 2598$_H$	(Prohibited area)				
FFE6 259C$_H$	(Prohibited area)				

Timer E Registers (FFE6 2000$_H$ to FFE6 2FFF$_H$) (6/8)

Address	b31 +3 b24	b23 +2 b16	b15 +1 b8	b7 +0 b0	Page
FFE6 25A0$_H$	(Prohibited area)				
to					
FFE6 25FC$_H$					
FFE6 2600$_H$	(Prohibited area)			SSTRE5	E5
FFE6 2604$_H$	(Prohibited area)			PSCRE5	
FFE6 2608$_H$	(Prohibited area)	RLDCRE5	(Prohibited area)	TCRE5	
FFE6 260C$_H$	(Prohibited area)	SOLVLE5	POECRE5		
FFE6 2610$_H$	TSCRE5		TSRE5		
FFE6 2614$_H$	TIERE5		(Prohibited area)	TOCRE5	
FFE6 2618$_H$	PSCCRE53	PSCCRE52	PSCCRE51	PSCCRE50	
FFE6 261C$_H$	(Prohibited area)				
FFE6 2620$_H$	(Prohibited area)				
FFE6 2624$_H$	TCNTE50				
FFE6 2628$_H$	CYLRE50				
FFE6 262C$_H$	DTRE50				
FFE6 2630$_H$	CRLDE50				
FFE6 2634$_H$	DRLDE50				
FFE6 2638$_H$	(Prohibited area)				
FFE6 263C$_H$	(Prohibited area)				
FFE6 2640$_H$	(Prohibited area)				
FFE6 2644$_H$	TCNTE51				
FFE6 2648$_H$	CYLRE51				
FFE6 264C$_H$	DTRE51				
FFE6 2650$_H$	CRLDE51				
FFE6 2654$_H$	DRLDE51				
FFE6 2658$_H$	(Prohibited area)				
FFE6 265C$_H$	(Prohibited area)				
FFE6 2660$_H$	(Prohibited area)				
FFE6 2664$_H$	TCNTE52				
FFE6 2668$_H$	CYLRE52				
FFE6 266C$_H$	DTRE52				
FFE6 2670$_H$	CRLDE52				
FFE6 2674$_H$	DRLDE52				
FFE6 2678$_H$	(Prohibited area)				
FFE6 267C$_H$	(Prohibited area)				
FFE6 2680$_H$	(Prohibited area)				
FFE6 2684$_H$	TCNTE53				
FFE6 2688$_H$	CYLRE53				
FFE6 268C$_H$	DTRE53				
FFE6 2690$_H$	CRLDE53				
FFE6 2694$_H$	DRLDE53				
FFE6 2698$_H$	(Prohibited area)				
FFE6 269C$_H$	(Prohibited area)				

Timer E Registers (FFE6 2000$_H$ to FFE6 2FFF$_H$) (7/8)

Address	b31 +3 b24	b23 +2 b16	b15 +1 b8	b7 +0 b0	Page
FFE6 26A0$_H$	(Prohibited area)				
to					
FFE6 26FC$_H$					
FFE6 2700$_H$	(Prohibited area)			SSTRE6	E6
FFE6 2704$_H$	(Prohibited area)			PSCRE6	
FFE6 2708$_H$	(Prohibited area)	RLDCRE6	(Prohibited area)	TCRE6	
FFE6 270C$_H$	(Prohibited area)	SOLVLE6	POECRE6		
FFE6 2710$_H$	TSCRE6		TSRE6		
FFE6 2714$_H$	TIERE6		(Prohibited area)	TOCRE6	
FFE6 2718$_H$	PSCCRE63	PSCCRE62	PSCCRE61	PSCCRE60	
FFE6 271C$_H$	(Prohibited area)				
FFE6 2720$_H$	(Prohibited area)				
FFE6 2724$_H$	TCNTE60				
FFE6 2728$_H$	CYLRE60				
FFE6 272C$_H$	DTRE60				
FFE6 2730$_H$	CRLDE60				
FFE6 2734$_H$	DRLDE60				
FFE6 2738$_H$	(Prohibited area)				
FFE6 273C$_H$	(Prohibited area)				
FFE6 2740$_H$	(Prohibited area)				
FFE6 2744$_H$	TCNTE61				
FFE6 2748$_H$	CYLRE61				
FFE6 274C$_H$	DTRE61				
FFE6 2750$_H$	CRLDE61				
FFE6 2754$_H$	DRLDE61				
FFE6 2758$_H$	(Prohibited area)				
FFE6 275C$_H$	(Prohibited area)				
FFE6 2760$_H$	(Prohibited area)				
FFE6 2764$_H$	TCNTE62				
FFE6 2768$_H$	CYLRE62				
FFE6 276C$_H$	DTRE62				
FFE6 2770$_H$	CRLDE62				
FFE6 2774$_H$	DRLDE62				
FFE6 2778$_H$	(Prohibited area)				
FFE6 277C$_H$	(Prohibited area)				
FFE6 2780$_H$	(Prohibited area)				
FFE6 2784$_H$	TCNTE63				
FFE6 2788$_H$	CYLRE63				
FFE6 278C$_H$	DTRE63				
FFE6 2790$_H$	CRLDE63				
FFE6 2794$_H$	DRLDE63				
FFE6 2798$_H$	(Prohibited area)				
FFE6 279C$_H$	(Prohibited area)				

Timer E Registers (FFE6 2000$_H$ to FFE6 2FFF$_H$) (8/8)

Address	b31 +3 b24	b23 +2 b16	b15 +1 b8	b7 +0 b0	Page
FFE6 27A0$_H$ to FFE6 2FFC$_H$	(Prohibited area)				

Timer F Registers (FFE6 3000$_H$ to FFE6 38FC$_H$) (1/7)

Address	b31 +3 b24	b23 +2 b16	b15 +1 b8	b7 +0 b0	Page
FFE6 3000$_H$	(Prohibited area)		TSTRF		Entire F
FFE6 3004$_H$	(Prohibited area)		NCMCR1F		
FFE6 3008$_H$	(Prohibited area)		(Prohibited area) NCMCR2F		
FFE6 300C$_H$	(Prohibited area)		NCCRF		
FFE6 3010$_H$	(Prohibited area)		PVFCRF		
FFE6 3014$_H$ to FFE6 3038$_H$	(Prohibited area)				
FFE6 303C$_H$	(Prohibited area)				
FFE6 3040$_H$	(Prohibited area)	TCR2F0	(Prohibited area)	TCR1F0	F0
FFE6 3044$_H$	TIERF0	TSCRF0	TSRF0	BKCRF0	
FFE6 3048$_H$	NCRFA0		NCNTFA0		
FFE6 304C$_H$	NCRFB0		NCNTFB0		
FFE6 3050$_H$	ECNTAF0				
FFE6 3054$_H$	GRAF0				
FFE6 3058$_H$	(Prohibited area)		ECNTBF0		
FFE6 305C$_H$	(Prohibited area)		GRBF0		
FFE6 3060$_H$	ECNTCF0				
FFE6 3064$_H$	GRCF0 (when ARSWCF0 = 0) BGRCF0 (when ARSWCF0 = 1)				
FFE6 3068$_H$	GRDF0				
FFE6 306C$_H$	CDRF0				
FFE6 3070$_H$	(Prohibited area)				
FFE6 3074$_H$	(Prohibited area)				
FFE6 3078$_H$	(Prohibited area)				
FFE6 307C$_H$	(Prohibited area)				
FFE6 3080$_H$	(Prohibited area)	TCR2F1	(Prohibited area)	TCR1F1	F1
FFE6 3084$_H$	TIERF1	TSCRF1	TSRF1	BKCRF1	
FFE6 3088$_H$	NCRFA1		NCNTFA1		
FFE6 308C$_H$	NCRFB1		NCNTFB1		
FFE6 3090$_H$	ECNTAF1				
FFE6 3094$_H$	GRAF1				
FFE6 3098$_H$	(Prohibited area)		ECNTBF1		
FFE6 309C$_H$	(Prohibited area)		GRBF1		
FFE6 30A0$_H$	ECNTCF1				
FFE6 30A4$_H$	GRCF1 (when ARSWCF1 = 0) BGRCF1 (when ARSWCF1 = 1)				

Timer F Registers (FFE6 3000$_H$ to FFE6 38FC$_H$) (2/7)

Address	b31 +3 b24	b23 +2 b16	b15 +1 b8	b7 +0 b0	Page
FFE6 30A8$_H$	GRDF1				
FFE6 30AC$_H$	CDRF1				
FFE6 30B0$_H$	(Prohibited area)				
FFE6 30B4$_H$	(Prohibited area)				
FFE6 30B8$_H$	(Prohibited area)				
FFE6 30BC$_H$	(Prohibited area)				
FFE6 30C0$_H$	(Prohibited area)	TCR2F2	(Prohibited area)	TCR1F2	F2
FFE6 30C4$_H$	TIERF2	TSCRF2	TSRF2	BKCRF2	
FFE6 30C8$_H$	NCRFA2		NCNTFA2		
FFE6 30CC$_H$	NCRFB2		NCNTFB2		
FFE6 30D0$_H$	ECNTAF2				
FFE6 30D4$_H$	GRAF2				
FFE6 30D8$_H$	(Prohibited area)		ECNTBF2		
FFE6 30DC$_H$	(Prohibited area)		GRBF2		
FFE6 30E0$_H$	ECNTCF2				
FFE6 30E4$_H$	GRCF2 (when ARSWCF2 = 0) BGRCF2 (when ARSWCF2 = 1)				
FFE6 30E8$_H$	GRDF2				
FFE6 30EC$_H$	CDRF2				
FFE6 30F0$_H$	(Prohibited area)				
FFE6 30F4$_H$	(Prohibited area)				
FFE6 30F8$_H$	(Prohibited area)				
FFE6 30FC$_H$	(Prohibited area)				
FFE6 3100$_H$	(Prohibited area)	TCR2F3	(Prohibited area)	TCR1F3	F3
FFE6 3104$_H$	TIERF3	TSCRF3	TSRF3	BKCRF3	
FFE6 3108$_H$	NCRFA3		NCNTFA3		
FFE6 310C$_H$	(Prohibited area)				
FFE6 3110$_H$	ECNTAF3				
FFE6 3114$_H$	GRAF3 (when ARSWAF3 = 0) BGRAF3 (when ARSWAF3 = 1)				
FFE6 3118$_H$	(Prohibited area)		ECNTBF3		
FFE6 311C$_H$	(Prohibited area)		GRBF3		
FFE6 3120$_H$	ECNTCF3				
FFE6 3124$_H$	GRCF3 (when ARSWCF3 = 0) BGRCF3 (when ARSWCF3 = 1)				
FFE6 3128$_H$	GRDF3 (when ARSWDF3 = 0) BGRDF3 (when ARSWDF3 = 1)				
FFE6 312C$_H$	CDRF3				
FFE6 3130$_H$	(Prohibited area)				
FFE6 3134$_H$	(Prohibited area)				
FFE6 3138$_H$	(Prohibited area)				
FFE6 313C$_H$	(Prohibited area)				
FFE6 3140$_H$	(Prohibited area)	TCR2F4	(Prohibited area)	TCR1F4	F4
FFE6 3144$_H$	TIERF4	TSCRF4	TSRF4	BKCRF4	
FFE6 3148$_H$	NCRFA4		NCNTFA4		

Timer F Registers (FFE6 3000$_H$ to FFE6 38FC$_H$) (3/7)

Address	b31 +3 b24	b23 +2 b16	b15 +1 b8	b7 +0 b0	Page
FFE6 314C$_H$	(Prohibited area)				
FFE6 3150$_H$	ECNTAF4				
FFE6 3154$_H$	GRAF4 (when ARSWAF4 = 0) BGRAF4 (when ARSWAF4 = 1)				
FFE6 3158$_H$	(Prohibited area)		ECNTBF4		
FFE6 315C$_H$	(Prohibited area)		GRBF4		
FFE6 3160$_H$	ECNTCF4				
FFE6 3164$_H$	GRCF4 (when ARSWCF4 = 0) BGRCF4 (when ARSWCF4 = 1)				
FFE6 3168H	GRDF4 (when ARSWDF4 = 0) BGRDF4 (when ARSWDF4 = 1)				
FFE6 316C$_H$	CDRF4				
FFE6 3170$_H$	(Prohibited area)				
FFE6 3174$_H$	(Prohibited area)				
FFE6 3178$_H$	(Prohibited area)				
FFE6 317C$_H$	(Prohibited area)				
FFE6 3180$_H$	(Prohibited area)	TCR2F5	(Prohibited area)	TCR1F5	F5
FFE6 3184$_H$	TIERF5	TSCRF5	TSRF5	BKCRF5	
FFE6 3188$_H$	NCRFA5		NCNTFA5		
FFE6 318C$_H$	(Prohibited area)				
FFE6 3190$_H$	ECNTAF5				
FFE6 3194$_H$	GRAF5 (when ARSWAF5 = 0) BGRAF5 (when ARSWAF5 = 1)				
FFE6 3198$_H$	(Prohibited area)		ECNTBF5		
FFE6 319C$_H$	(Prohibited area)		GRBF5		
FFE6 31A0$_H$	ECNTCF5				
FFE6 31A4$_H$	GRCF5 (when ARSWCF5 = 0) BGRCF5 (when ARSWCF5 = 1)				
FFE6 31A8$_H$	GRDF5 (when ARSWDF5 = 0) BGRDF5 (when ARSWDF5 = 1)				
FFE6 31AC$_H$	CDRF5				
FFE6 31B0$_H$	(Prohibited area)				
FFE6 31B4$_H$	(Prohibited area)				
FFE6 31B8$_H$	(Prohibited area)				
FFE6 31BC$_H$	(Prohibited area)				
FFE6 31C0$_H$	(Prohibited area)	TCR2F6	(Prohibited area)	TCR1F6	F6
FFE6 31C4$_H$	TIERF6	TSCRF6	TSRF6	BKCRF6	
FFE6 31C8$_H$	NCRFA6		NCNTFA6		
FFE6 31CC$_H$	(Prohibited area)				
FFE6 31D0$_H$	ECNTAF6				
FFE6 31D4$_H$	GRAF6 (when ARSWAF6 = 0) BGRAF6 (when ARSWAF6 = 1)				
FFE6 31D8$_H$	(Prohibited area)		ECNTBF6		
FFE6 31DC$_H$	(Prohibited area)		GRBF6		
FFE6 31E0$_H$	ECNTCF6				
FFE6 31E4$_H$	GRCF6 (when ARSWCF6 = 0) BGRCF6 (when ARSWCF6 = 1)				

Timer F Registers (FFE6 3000$_H$ to FFE6 38FC$_H$) (4/7)

Address	b31 +3 b24	b23 +2 b16	b15 +1 b8	b7 +0 b0	Page
FFE6 31E8$_H$	GRDF6 (when ARSWDF6 = 0) BGRDF6 (when ARSWDF6 = 1)				
FFE6 31EC$_H$	CDRF6				
FFE6 31F0$_H$	(Prohibited area)				
FFE6 31F4$_H$	(Prohibited area)				
FFE6 31F8$_H$	(Prohibited area)				
FFE6 31FC$_H$	(Prohibited area)				
FFE6 3200$_H$	(Prohibited area)	TCR2F7	(Prohibited area)	TCR1F7	F7
FFE6 3204$_H$	TIERF7	TSCRF7	TSRF7	BKCRF7	
FFE6 3208$_H$	NCRFA7		NCNTFA7		
FFE6 320C$_H$	(Prohibited area)				
FFE6 3210$_H$	ECNTAF7				
FFE6 3214$_H$	GRAF7 (when ARSWAF7 = 0) BGRAF7 (when ARSWAF7 = 1)				
FFE6 3218$_H$	(Prohibited area)		ECNTBF7		
FFE6 321C$_H$	(Prohibited area)		GRBF7		
FFE6 3220$_H$	ECNTCF7				
FFE6 3224$_H$	GRCF7 (when ARSWCF7 = 0) BGRCF7 (when ARSWCF7 = 1)				
FFE6 3228$_H$	GRDF7 (when ARSWDF7 = 0) BGRDF7 (when ARSWDF7 = 1)				
FFE6 322C$_H$	CDRF7				
FFE6 3230$_H$	(Prohibited area)				
FFE6 3234$_H$	(Prohibited area)				
FFE6 3238$_H$	(Prohibited area)				
FFE6 323C$_H$	(Prohibited area)				
FFE6 3240$_H$	(Prohibited area)	TCR2F8	(Prohibited area)	TCR1F8	F8
FFE6 3244$_H$	TIERF8	TSCRF8	TSRF8	BKCRF8	
FFE6 3248$_H$	NCRFA8		NCNTFA8		
FFE6 324C$_H$	(Prohibited area)				
FFE6 3250$_H$	ECNTAF8				
FFE6 3254$_H$	GRAF8 (when ARSWAF8 = 0) BGRAF8 (when ARSWAF8 = 1)				
FFE6 3258$_H$	(Prohibited area)		ECNTBF8		
FFE6 325C$_H$	(Prohibited area)		GRBF8		
FFE6 3260$_H$	ECNTCF8				
FFE6 3264$_H$	GRCF8 (when ARSWCF8 = 0) BGRCF8 (when ARSWCF8 = 1)				
FFE6 3268$_H$	GRDF8 (when ARSWDF8 = 0) BGRDF8 (when ARSWDF8 = 1)				
FFE6 326C$_H$	CDRF8				
FFE6 3270$_H$	(Prohibited area)				
FFE6 3274$_H$	(Prohibited area)				
FFE6 3278$_H$	(Prohibited area)				
FFE6 327C$_H$	(Prohibited area)				
FFE6 3280$_H$	(Prohibited area)	TCR2F9	(Prohibited area)	TCR1F9	F9

Timer F Registers (FFE6 3000$_H$ to FFE6 38FC$_H$) (5/7)

Address	b31 +3 b24	b23 +2 b16	b15 +1 b8	b7 +0 b0	Page
FFE6 3284$_H$	TIERF9	TSCRF9	TSRF9	BKCRF9	
FFE6 3288$_H$	NCRFA9		NCNTFA9		
FFE6 328C$_H$	(Prohibited area)				
FFE6 3290$_H$	ECNTAF9				
FFE6 3294$_H$	GRAF9 (when ARSWAF9 = 0) BGRAF9 (when ARSWAF9 = 1)				
FFE6 3298$_H$	(Prohibited area)		ECNTBF9		
FFE6 329C$_H$	(Prohibited area)		GRBF9		
FFE6 32A0$_H$	ECNTCF9				
FFE6 32A4$_H$	GRCF9 (when ARSWCF9 = 0) BGRCF9 (when ARSWCF9 = 1)				
FFE6 32A8$_H$	GRDF9 (when ARSWDF9 = 0) BGRDF9 (when ARSWDF9 = 1)				
FFE6 32AC$_H$	CDRF9				
FFE6 32B0$_H$	(Prohibited area)				
FFE6 32B4$_H$	(Prohibited area)				
FFE6 32B8$_H$	(Prohibited area)				
FFE6 32BC$_H$	(Prohibited area)				
FFE6 32C0$_H$	(Prohibited area)	TCR2F10	(Prohibited area)	TCR1F10	F10
FFE6 32C4$_H$	TIERF10	TSCRF10	TSRF10	BKCRF10	
FFE6 32C8$_H$	NCRFA10		NCNTFA10		
FFE6 32CC$_H$	(Prohibited area)				
FFE6 32D0$_H$	ECNTAF10				
FFE6 32D4$_H$	GRAF10 (when ARSWAF10 = 0) BGRAF10 (when ARSWAF10 = 1)				
FFE6 32D8$_H$	(Prohibited area)		ECNTBF10		
FFE6 32DC$_H$	(Prohibited area)		GRBF10		
FFE6 32E0$_H$	ECNTCF10				
FFE6 32E4$_H$	GRCF10 (when ARSWCF10 = 0) BGRCF10 (when ARSWCF10 = 1)				
FFE6 32E8$_H$	GRDF10 (when ARSWDF10 = 0) BGRDF10 (when ARSWDF10 = 1)				
FFE6 32EC$_H$	CDRF10				
FFE6 32F0$_H$	(Prohibited area)				
FFE6 32F4$_H$	(Prohibited area)				
FFE6 32F8$_H$	(Prohibited area)				
FFE6 32FC$_H$	(Prohibited area)				
FFE6 3300$_H$	(Prohibited area)	TCR2F11	(Prohibited area)	TCR1F11	F11
FFE6 3304$_H$	TIERF11	TSCRF11	TSRF11	BKCRF11	
FFE6 3308$_H$	NCRFA11		NCNTFA11		
FFE6 330C$_H$	(Prohibited area)				
FFE6 3310$_H$	ECNTAF11				
FFE6 3314$_H$	GRAF11 (when ARSWAF11 = 0) BGRAF11 (when ARSWAF11 = 1)				
FFE6 3318$_H$	(Prohibited area)		ECNTBF11		
FFE6 331C$_H$	(Prohibited area)		GRBF11		

Timer F Registers (FFE6 3000$_H$ to FFE6 38FC$_H$) (6/7)

Address	b31 +3 b24	b23 +2 b16	b15 +1 b8	b7 +0 b0	Page
FFE6 3320$_H$	ECNTCF11				
FFE6 3324$_H$	GRCF11 (when ARSWCF11 = 0) BGRCF11 (when ARSWCF11 = 1)				
FFE6 3328$_H$	GRDF11 (when ARSWDF11 = 0) BGRDF11 (when ARSWDF11 = 1)				
FFE6 332C$_H$	CDRF11				
FFE6 3330$_H$	(Prohibited area)				
FFE6 3334$_H$	(Prohibited area)				
FFE6 3338$_H$	(Prohibited area)				
FFE6 333C$_H$	(Prohibited area)				
FFE6 3340$_H$	(Prohibited area)	TCR2F12	(Prohibited area)	TCR1F12	F12
FFE6 3344$_H$	TIERF12	TSCRF12	TSRF12	BKCRF12	
FFE6 3348$_H$	NCRFA12		NCNTFA12		
FFE6 334C$_H$	(Prohibited area)				
FFE6 3350$_H$	ECNTAF12				
FFE6 3354$_H$	GRAF12 (when ARSWAF12 = 0) BGRAF12 (when ARSWAF12 = 1)				
FFE6 3358$_H$	(Prohibited area)		ECNTBF12		
FFE6 335C$_H$	(Prohibited area)		GRBF12		
FFE6 3360$_H$	ECNTCF12				
FFE6 3364$_H$	GRCF12 (when ARSWCF12 = 0) BGRCF12 (when ARSWCF12 = 1)				
FFE6 3368$_H$	GRDF12 (when ARSWDF12 = 0) BGRDF12 (when ARSWDF12 = 1)				
FFE6 336C$_H$	CDRF12				
FFE6 3370$_H$	(Prohibited area)				
FFE6 3374$_H$	(Prohibited area)				
FFE6 3378$_H$	(Prohibited area)				
FFE6 337C$_H$	(Prohibited area)				
FFE6 3380$_H$	(Prohibited area)	TCR2F13	(Prohibited area)	TCR1F13	F13
FFE6 3384$_H$	TIERF13	TSCRF13	TSRF13	BKCRF13	
FFE6 3388$_H$	NCRFA13		NCNTFA13		
FFE6 338C$_H$	(Prohibited area)				
FFE6 3390$_H$	ECNTAF13				
FFE6 3394$_H$	GRAF13 (when ARSWAF13 = 0) BGRAF13 (when ARSWAF13 = 1)				
FFE6 3398$_H$	(Prohibited area)		ECNTBF13		
FFE6 339C$_H$	(Prohibited area)		GRBF13		
FFE6 33A0$_H$	ECNTCF13				
FFE6 33A4$_H$	GRCF13 (when ARSWCF13 = 0) BGRCF13 (when ARSWCF13 = 1)				
FFE6 33A8$_H$	GRDF13 (when ARSWDF13 = 0) BGRDF13 (when ARSWDF13 = 1)				
FFE6 33AC$_H$	CDRF13				
FFE6 33B0$_H$	(Prohibited area)				
FFE6 33B4$_H$	(Prohibited area)				
FFE6 33B8$_H$	(Prohibited area)				

Timer F Registers (FFE6 3000$_H$ to FFE6 38FC$_H$) (7/7)

Address	b31 +3 b24	b23 +2 b16	b15 +1 b8	b7 +0 b0	Page
FFE6 33BC$_H$	(Prohibited area)				
FFE6 33C0$_H$	(Prohibited area)	TCR2F14	(Prohibited area)	TCR1F14	F14
FFE6 33C4$_H$	TIERF14	TSCRF14	TSRF14	BKCRF14	
FFE6 33C8$_H$	NCRFA14		NCNTFA14		
FFE6 33CC$_H$	(Prohibited area)				
FFE6 33D0$_H$	ECNTAF14				
FFE6 33D4$_H$	GRAF14 (when ARSWAF14 = 0) BGRAF14 (when ARSWAF14 = 1)				
FFE6 33D8$_H$	(Prohibited area)		ECNTBF14		
FFE6 33DC$_H$	(Prohibited area)		GRBF14		
FFE6 33E0$_H$	ECNTCF14				
FFE6 33E4$_H$	GRCF14 (when ARSWCF14 = 0) BGRCF14 (when ARSWCF14 = 1)				
FFE6 33E8$_H$	GRDF14 (when ARSWDF14 = 0) BGRDF14 (when ARSWDF14 = 1)				
FFE6 33EC$_H$	CDRF14				
FFE6 33F0$_H$	(Prohibited area)				
FFE6 33F4$_H$	(Prohibited area)				
FFE6 33F8$_H$	(Prohibited area)				
FFE6 33FC$_H$	(Prohibited area)				
FFE6 3400$_H$	(Prohibited area)	TCR2F15	(Prohibited area)	TCR1F15	F15
FFE6 3404$_H$	TIERF15	TSCRF15	TSRF15	BKCRF15	
FFE6 3408$_H$	NCRFA15		NCNTFA15		
FFE6 340C$_H$	(Prohibited area)				
FFE6 3410$_H$	ECNTAF15				
FFE6 3414$_H$	GRAF15 (when ARSWAF15 = 0) BGRAF15 (when ARSWAF15 = 1)				
FFE6 3418$_H$	(Prohibited area)		ECNTBF15		
FFE6 341C$_H$	(Prohibited area)		GRBF15		
FFE6 3420$_H$	ECNTCF15				
FFE6 3424$_H$	GRCF15 (when ARSWCF15 = 0) BGRCF15 (when ARSWCF15 = 1)				
FFE6 3428$_H$	GRDF15 (when ARSWDF15 = 0) BGRDF15 (when ARSWDF15 = 1)				
FFE6 342C$_H$	CDRF15				
FFE6 3430$_H$ to FFE6 38FC$_H$	(Prohibited area)				

Timer G Registers (FFE6 3900$_H$ to FFE6 3BFF$_H$) (1/2)

Address	b31 +3 b24	b23 +2 b16	b15 +1 b8	b7 +0 b0	Page
FFE6 3900$_H$	(Prohibited area)		TSTRG		Entire G
FFE6 3904$_H$	(Prohibited area)				
FFE6 3908$_H$	(Prohibited area)				
FFE6 390C$_H$	(Prohibited area)				
FFE6 3910$_H$	TSCRG0	TSRG0	(Prohibited area)	TCRG0	G0
FFE6 3914$_H$	TCNTG0				
FFE6 3918$_H$	OCRG0				
FFE6 391C$_H$	(Prohibited area)				
FFE6 3920$_H$	TSCRG1	TSRG1	(Prohibited area)	TCRG1	G1
FFE6 3924$_H$	TCNTG1				
FFE6 3928$_H$	OCRG1				
FFE6 392C$_H$	(Prohibited area)				
FFE6 3930$_H$	TSCRG2	TSRG2	(Prohibited area)	TCRG2	G2
FFE6 3934$_H$	TCNTG2				
FFE6 3938$_H$	OCRG2				
FFE6 393C$_H$	(Prohibited area)				
FFE6 3940$_H$	TSCRG3	TSRG3	(Prohibited area)	TCRG3	G3
FFE6 3944$_H$	TCNTG3				
FFE6 3948$_H$	OCRG3				
FFE6 394C$_H$	(Prohibited area)				
FFE6 3950$_H$	TSCRG4	TSRG4	(Prohibited area)	TCRG4	G4
FFE6 3954$_H$	TCNTG4				
FFE6 3958$_H$	OCRG4				
FFE6 395C$_H$	(Prohibited area)				
FFE6 3960$_H$	TSCRG5	TSRG5	(Prohibited area)	TCRG5	G5
FFE6 3964$_H$	TCNTG5				
FFE6 3968$_H$	OCRG5				
FFE6 396C$_H$	(Prohibited area)				
FFE6 3970$_H$	TSCRG6	TSRG6	(Prohibited area)	TCRG6	G6
FFE6 3974$_H$	TCNTG6				
FFE6 3978$_H$	OCRG6				
FFE6 397C$_H$	(Prohibited area)				
FFE6 3980$_H$	TSCRG7	TSRG7	(Prohibited area)	TCRG7	G7
FFE6 3984$_H$	TCNTG7				
FFE6 3988$_H$	OCRG7				
FFE6 398C$_H$	(Prohibited area)				
FFE6 3990$_H$	TSCRG8	TSRG8	(Prohibited area)	TCRG8	G8
FFE6 3994$_H$	TCNTG8				
FFE6 3998$_H$	OCRG8				
FFE6 399C$_H$	(Prohibited area)				
FFE6 39A0$_H$	TSCRG9	TSRG9	(Prohibited area)	TCRG9	G9
FFE6 39A4$_H$	TCNTG9				
FFE6 39A8$_H$	OCRG9				
FFE6 39AC$_H$	(Prohibited area)				

Timer G Registers (FFE6 3900$_H$ to FFE6 3BFF$_H$) (2/2)

Address	b31 +3 b24	b23 +2 b16	b15 +1 b8	b7 +0 b0	Page
FFE6 39B0$_H$	(Prohibited area)				
to					
FFE6 3BEC$_H$					
FFE6 3BF0$_H$	(Prohibited area)				
to					
FFE6 3BFC$_H$					

Timer H Register (FFE6 0100$_H$ to FFE6 01FF$_H$)

Address	b31 +3 b24	b23 +2 b16	b15 +1 b8	b7 +0 b0	Page
FFE6 0100$_H$	TSCRH	TSRH	(Prohibited area)	TCRH	H
FFE6 0104$_H$	OCR1H		TCNT1H		
FFE6 0108$_H$	TCNT2H				
FFE6 010C$_H$	(Prohibited area)				
FFE6 0110$_H$	(Prohibited area)				
to					
FFE6 01FC$_H$					

Timer J Registers (FFE6 3C00$_H$ to FFE6 3FFF$_H$) (1/2)

Address	b31 +3 b24	b23 +2 b16	b15 +1 b8	b7 +0 b0	Page
FFE6 3C00$_H$	(Prohibited area)			TSTRJ	Entire J
FFE6 3C04$_H$	(Prohibited area)				
FFE6 3C08$_H$	(Prohibited area)				
to					
FFE6 3C1C$_H$					
FFE6 3C20$_H$	TSCRJ0	TSRJ0	FCRJ0	TCRJ0	J0
FFE6 3C24$_H$	TCNTJ0				
FFE6 3C28$_H$	OCRJ0				
FFE6 3C2C$_H$	FIFOJ0				
FFE6 3C30$_H$	(Prohibited area)			FDNRJ0	
FFE6 3C34$_H$	NCRJ0		NCNTJ0		
FFE6 3C38$_H$	(Prohibited area)				
FFE6 3C3C$_H$	(Prohibited area)				
FFE6 3C40$_H$	TSCRJ1	TSRJ1	FCRJ1	TCRJ1	J1
FFE6 3C44$_H$	TCNTJ1				
FFE6 3C48$_H$	OCRJ1				
FFE6 3C4C$_H$	FIFOJ1				
FFE6 3C50$_H$	(Prohibited area)			FDNRJ1	
FFE6 3C54$_H$	NCRJ1		NCNTJ1		
FFE6 3C58$_H$	(Prohibited area)				
FFE6 3C5C$_H$	(Prohibited area)				
FFE6 3C60$_H$	TSCRJ2	TSRJ2	FCRJ2	TCRJ2	J2

Timer J Registers (FFE6 3C00$_H$ to FFE6 3FFF$_H$) (2/2)

Address	b31 +3 b24	b23 +2 b16	b15 +1 b8	b7 +0 b0	Page
FFE6 3C64$_H$	TCNTJ2				
FFE6 3C68$_H$	OCRJ2				
FFE6 3C6C$_H$	FIFOJ2				
FFE6 3C70$_H$	(Prohibited area)			FDNRJ2	
FFE6 3C74$_H$	NCRJ2		NCNTJ2		
FFE6 3C78$_H$	(Prohibited area)				
FFE6 3C7C$_H$	(Prohibited area)				
FFE6 3C80$_H$	TSCRJ3	TSRJ3	FCRJ3	TCRJ3	J3
FFE6 3C84$_H$	TCNTJ3				
FFE6 3C88$_H$	OCRJ3				
FFE6 3C8C$_H$	FIFOJ3				
FFE6 3C90$_H$	(Prohibited area)			FDNRJ3	
FFE6 3C94$_H$	NCRJ3		NCNTJ3		
FFE6 3C98$_H$	(Prohibited area)				
FFE6 3C9C$_H$	(Prohibited area)				
FFE6 3CA0$_H$	TSCRJ4	TSRJ4	FCRJ4	TCRJ4	J4
FFE6 3CA4$_H$	TCNTJ4				
FFE6 3CA8$_H$	OCRJ4				
FFE6 3CAC$_H$	FIFOJ4				
FFE6 3CB0$_H$	(Prohibited area)			FDNRJ4	
FFE6 3CB4$_H$	NCRJ4		NCNTJ4		
FFE6 3CB8$_H$	(Prohibited area)				
FFE6 3CBC$_H$	(Prohibited area)				
FFE6 3CC0$_H$	TSCRJ5	TSRJ5	FCRJ5	TCRJ5	J5
FFE6 3CC4$_H$	TCNTJ5				
FFE6 3CC8$_H$	OCRJ5				
FFE6 3CCC$_H$	FIFOJ5				
FFE6 3CD0$_H$	(Prohibited area)			FDNRJ5	
FFE6 3CD4$_H$	NCRJ5		NCNTJ5		
FFE6 3CD8$_H$	(Prohibited area)				
FFE6 3CDC$_H$	(Prohibited area)				
FFE6 3CE0$_H$	(Prohibited area)				
to					
FFE6 3FFF$_H$					

21.1.3 ATU-IV Input/Output Signals

Common Controller

Symbol	I/O	Number of Bits	Function
TCLKA	Input	1	Input pin for the external clock on signal line 4 of the clock bus
TCLKB	Input	1	Input pin for the external clock on signal line 5 of the clock bus

Prescaler

Not provided

Timer A

Symbol	I/O	Number of Bits	Function
TIA00 to TIA06	Input	7	Input pins for input-capture triggers for timer A channels

Timer B

Not provided

Timer C

Symbol	I/O	Number of Bits	Function
TIOC00 - 03 to TIOC70 - 73	Input	32	Input pins for input-capture triggers and output pins for output-compare signals of timer C (one each for channels 0 to 3 in subblocks C0 to C7)

Timer D

Symbol	I/O	Number of Bits	Function
TOD00A - 03A to TOD90A - 93A*1	Output	40	Output pins for compare-match signals of timer D (one each for channels 0 to 3 in subblocks D0 to D9)
TOD00B - 03B to TOD90B - 93B*1	Output	40	Output pins for one-shot pulses of timer D (one each for channels 0 to 3 in subblocks D0 to D9)

Note 1. The subblock D9 is used to activate the A/D and DFE and has no external input/output pin.

Timer E

Symbol	I/O	Number of Bits	Function
POE0 to POE5	Input	6	Timer E (subblocks E0 to E5) shutoff input * Shared with TIF0A to TIF2A and TIF0B to TIF2B
TOE00 - 03 to TOE60 - 63	Output	28	Output pins for PWM signals of timer E (one each for channels 0 to 3 in subblocks E0 to E6)

Timer F

Symbol	I/O	Number of Bits	Function
TIF0A to TIF2A TIF3 to TIF15	Input	16	Input pins for event signals for subblocks F0 to F15 of timer F TIF0A to TIF2A: Input pins for subblocks F0 to F2 TIF3 to TIF15: Input pins for subblocks F3 to F15
TIF0B to TIF2B	Input	3	Input pins for event signals for subblocks F0 to F2 of timer F

Timer G

Not provided

Timer H

Not provided

Timer J

Symbol	I/O	Number of Bits	Function
TIJ0 to TIJ5	Input	6	Input pins for input-capture triggers for subblocks of timer J

21.1.4 Clock Supply

The clock supply for ATU-IV is listed in the following table.

Clock for the Unit	Supply Clock Name
Pϕ, PCLK, pclk	CLKC_LSB (non-modulated low speed peripheral clock)

21.2 Common Controller

21.2.1 Operation

The common controller controls the ATU-IV module as a whole. For example, it enables and disables the prescalers and timer counters for timers A to J and controls the clock bus.

Clock Bus

The clock bus consists of seven signal lines used to distribute the source signals for counting (counter enabling signals) to the timer channels. The timer counters on each of the channels run in synchronization with the internal peripheral clock (Pϕ).

The following table shows the signals which are available for input on the clock bus.

Bit Number of Clock Bus	Input Signals
6	Timer B output (AGCKM2)
5	Output signal from timer B (angle clock) or external input clock B (TCLKB)
4	External input clock A (TCLKA)
3	Output signal from prescaler 3
2	Output signal from prescaler 2
1	Output signal from prescaler 1
0	Output signal from prescaler 0

21.2.2 Register Description of Common Controller

21.2.2.1 ATUENR — ATU-IV Master Enable Register

Access: 8-bit and 16-bit accessible

Address: FFE6 0000_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TJE	THE	TGE	TFE	TEE	TDE	TCE	TBE	TAE	PSCE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.2 ATUENR Register Contents

Bit Position	Bit Name	Function
15 to 10	—	These bits are not used. Fix these bits to 0.
9	TJE	Timer J Enable 0: Timer J counter operation disabled 1: Timer J counter operation enabled
8	THE	Timer H Enable 0: Timer H counter operation disabled 1: Timer H counter operation enabled
7	TGE	Timer G Enable 0: Timer G counter operation disabled 1: Timer G counter operation enabled
6	TFE	Timer F Enable 0: Timer F counter operation disabled 1: Timer F counter operation enabled
5	TEE	Timer E Enable 0: Timer E counter operation disabled 1: Timer E counter operation enabled
4	TDE	Timer D Enable 0: Timer D counter operation disabled 1: Timer D counter operation enabled
3	TCE	Timer C Enable 0: Timer C counter operation disabled 1: Timer C counter operation enabled
2	TBE	Timer B Enable 0: Timer B counter operation disabled 1: Timer B counter operation enabled
1	TAE	Timer A Enable 0: Timer A counter operation disabled 1: Timer A counter operation enabled
0	PSCE	Prescaler Enable 0: Prescaler counter operation disabled 1: Prescaler counter operation enabled

ATUENR is a 16-bit readable/writable register. This register is used to enable and disable the prescalers and the individual blocks in ATU-IV. Setting an enable bit to 1 enables the corresponding block. Clearing the bit to 0 disables the corresponding block. Even when the enable bit is cleared to 0, the registers of these blocks remain accessible.

Timers can be synchronized by simultaneously setting multiple bits to 1.

(1) TJE — Timer J Enable

Enables and disables counter operation of timer J.

When this bit is set to 1, the timer J counter starts working. When this bit is cleared to 0, the timer J counter stops working. When the counter is disabled, its value is retained. When this bit is again set to 1, the counter resumes counting from the retained value.

However, the corresponding bit in the timer J start register must also be set to 1 to enable the operation of either of the subblock counters.

(2) THE — Timer H Enable

Enables and disables counter operation of timer H.

When this bit is set to 1, the timer H counter starts working. When this bit is cleared to 0, the timer H counter stops working. When the counter is disabled, its value is retained. When this bit is again set to 1, the counter resumes counting from the retained value.

(3) TGE — Timer G Enable

Enables and disables counter operation of timer G.

When this bit is set to 1, the timer G counter starts working. When this bit is cleared to 0, the timer G counter stops working. When the counter is disabled, its value is retained. When this bit is again set to 1, the counter resumes counting from the retained value.

However, the corresponding bit in the timer G start register must also be set to 1 to enable the operation of either of the subblock counters.

(4) TFE — Timer F Enable

Enables and disables counter operation of timer F.

When this bit is set to 1, the timer F counter starts working. When this bit is cleared to 0, the timer F counter stops working. When the counter is disabled, its value is retained. When this bit is again set to 1, the counter resumes counting from the retained value.

However, the corresponding bit in the timer F start register must also be set to 1 to enable the operation of either of the subblock counters.

(5) TEE — Timer E Enable

Enables and disables counter operation of timer E.

When this bit is set to 1, the timer E counter starts working. When this bit is cleared to 0, the timer E counter stops working. When the counter is disabled, its value is retained. When this bit is again set to 1, the counter resumes counting from the retained value.

However, the corresponding bit in the timer E start register must also be set to 1 to enable the operation of either of the subblock counters.

(6) TDE — Timer D Enable

Enables and disables counter operation of timer D.

When this bit is set to 1, the timer D counter starts working. When this bit is cleared to 0, the timer D counter stops working. When the counter is disabled, its value is retained. When this bit is again set to 1, the counter resumes counting from the retained value.

However, the corresponding bit in the timer D start register must also be set to 1 to enable the operation of either of the subblock counters.

(7) TCE — Timer C Enable

Enables and disables counter operation of timer C.

When this bit is set to 1, the timer C counter starts working. When this bit is cleared to 0, the timer C counter stops working. When the counter is disabled, its value is retained. When this bit is again set to 1, the counter resumes counting from the retained value.

However, the corresponding bit in the timer C start register must also be set to 1 to enable the operation of either of the subblock counters.

(8) TBE — Timer B Enable

Enables and disables counter operation of timer B.

When this bit is set to 1, the timer B counter starts working. When this bit is cleared to 0, the timer B counter stops working. When the counter is disabled, its value is retained. When this bit is again set to 1, the counter resumes counting from the retained value.

(9) TAE — Timer A Enable

Enables and disables counter operation of timer A.

When this bit is set to 1, the timer A counter starts working. When this bit is cleared to 0, the timer A counter stops working. When the counter is disabled, its value is retained. When this bit is again set to 1, the counter resumes counting from the retained value.

(10) PSCE — Prescaler Enable

Enables and disables the prescaler counters.

When this bit is set to 1, the prescaler counter starts working. When this bit is cleared to 0, the prescaler counter stops working. When the prescaler counters are disabled, the counter values are retained. Once the bit is set to 1 again, the counter resumes counting from the retained value.

21.2.2.2 CBCNT — Clock Bus Control Register

Access: 8-bit accessible

Address: FFE6 0002_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	CB4EG[1:0]		—	CB5SEL	CB5EG[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R	R/W	R/W	R/W

Table 21.3 CBCNT Register Contents

Bit Position	Bit Name	Function
7, 6	—	These bits are not used. Fix these bits to 0.
5, 4	CB4EG[1:0]	Clock Bus 4 Edge Select These bits specify the edge sense for external input clock (TCLKA) to be output on signal line 4 of the clock bus. 00: Neither edge of the external clock is sensed 01: Rising edges of the external clock are sensed 10: Falling edges of the external clock are sensed 11: Both rising and falling edges of the external clock are sensed
3	—	This bit is not used. Fix this bit to 0.
2	CB5SEL	Clock Bus 5 Source Select This bit specifies the clock to be output on signal line 5 of the clock bus. 0: External input clock B (TCLKB) 1: Multiplied-and-corrected clock output by Timer B
1, 0	CB5EG[1:0]	Clock Bus 5 Edge Select These bits specify the edge sense for external input clock to be output on signal line 5 of the clock bus when CB5S is cleared to 0. 00: Neither edge of the external clock is sensed 01: Rising edges of the external clock are sensed 10: Falling edges of the external clock are sensed 11: Both rising and falling edges of the external clock are sensed

CBCNT is an 8-bit readable/writable register that selects the source of the clock signal to be supplied on signal line 5 of the clock bus and the effective edge of the external clock signal (only applies to line 5 when the externally input clock is selected).

(1) CB4EG[1:0] — Clock Bus 4 Edge Select

These bits select the edge sense for externally input clock A (TCLKA). The clock signal is output on signal line 4 of the clock bus. Counters for which signal line 4 of the clock bus has been selected as the source for counting count on the edge selected by these bits.

(2) CB5SEL — Clock Bus 5 Source Select

Selects the source of the clock to be output on signal line 5 of the clock bus.

(3) CB5EG[1:0] — Clock Bus 5 Edge Select

These bits select the edge sense for external input clock B (TCLKB). The clock signal is output on signal line 5 of the clock bus. Counters for which signal line 5 of the clock bus has been selected as the source for counting count on the edge selected by these bits.

The setting of these bits is only valid when the TCLKB signal is selected as the source for line 5 of the clock bus. When the angle clock is selected as the source for line 5 of the clock bus, the setting of these bits is ignored.

21.2.2.3 NCMR — Noise Cancellation Mode Register

Access: 8-bit accessible

Address: FFE6 0003_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	NCCSEL	—	—	NCMSEL	NCMJ	NCMF	NCMC	NCMA
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W

Table 21.4 NCMR Register Contents

Bit Position	Bit Name	Function
7	NCCSEL	Noise Canceler Counter Clock Select This bit selects the clock for counting by the noise cancelers. 0: PCLK divided by 128 is used as the counter clock 1: PCLK is used as the counter clock
6, 5	—	These bits are not used. Fix these bits to 0.
4	NCMSEL	Noise Cancellation Mode Select This bit selects minimum time-at-level cancellation mode or level accumulation cancellation mode. 0: Minimum time-at-level cancellation mode 1: Level accumulation cancellation mode
3	NCMJ	Timer J Noise Cancellation Mode This bit specifies the operating mode of the timer J noise canceler. 0: Premature-transition cancellation mode 1: Minimum time-at-level cancellation mode (when NCMSEL = 0) Level accumulation cancellation mode (when NCMSEL = 1)
2	NCMF	Timer F Noise Cancellation Mode This bit specifies the operating mode of the timer F noise canceler. 0: Premature-transition cancellation mode 1: Minimum time-at-level cancellation mode (when NCMSEL = 0) Level accumulation cancellation mode (when NCMSEL = 1)
1	NCMC	Timer C Noise Cancellation Mode This bit specifies the operating mode of the timer C noise canceler. 0: Premature-transition cancellation mode 1: Minimum time-at-level cancellation mode (when NCMSEL = 0) Level accumulation cancellation mode (when NCMSEL = 1)
0	NCMA	Timer A Noise Cancellation Mode This bit specifies the operating mode of the timer A noise canceler. 0: Premature-transition cancellation mode 1: Minimum time-at-level cancellation mode (when NCMSEL = 0) Level accumulation cancellation mode (when NCMSEL = 1)

NCMR is an 8-bit readable/writable register that selects the mode and clock to drive the counter for of the noise canceler in each of timers A, C, F, and J.

In premature-transition cancellation mode, subsequent changes to the input signal level are ignored if they come within a given period of a detected change. That is, level changes within a certain period of an initial one are treated as noise.

In minimum time-at-level cancellation mode, the first and subsequent level changes are ignored unless the input signal level remains the same over a given period. Level changes occurring within a shorter period are considered to indicate an unstable signal, and such signals are treated as noise.

In level accumulation cancellation mode, input signal levels are accumulated and the input level is regarded to have reached 0 or 1 when the accumulation result becomes 0 or the specified value.

The period is set by noise canceler registers in each of the applicable blocks (i.e. in timers A, C, F, and J) and is counted by a noise canceler counter.

Outline of noise canceling operation (example of TIA00 input signal of timer A) is shown in **Figure 21.1** (in premature-transition cancellation mode), **Figure 21.2** (in minimum time-at-level cancellation mode), and **Figure 21.3** (in level accumulation cancellation mode).

Each timer (A, C, F, and J) detects the edge of a signal that has passed through the noise canceler. **Figure 21.1**, **Figure 21.2**, and **Figure 21.3** show an example of detecting the rising edge of a signal that has passed through the noise canceler.

(1) NCCSEL — Noise Canceler Counter Clock Select

Selects the clock for counting by the noise cancelers. Either of the internal peripheral clock (PCLK) divided by 128 or the internal peripheral clock can be selected. The default setting is the clock divided by 128. The same counter clock must be used for all timers other than timer A. In the case of timer A, the clock signal on clock-bus line 5 is also available. For setting the counter clock, see the description of the TIOR2A register in timer A.

(2) NCMSEL — Noise Cancellation Mode Select

Noise cancellation mode when each noise cancellation mode bit (NCMA, NCMC, NCMF, or NCMJ) is set to 1 can be set to premature-transition cancellation mode or level accumulation cancellation mode.

(3) NCMJ — Noise Canceler Counter Clock Select

Selects the noise cancellation mode for timer J. A different operating mode cannot be set for each channel of timer J.

(4) NCMF — Timer F Noise Cancellation Mode

Selects the noise cancellation mode for timer F. A different operating mode cannot be set for each channel of timer F. The mode of operation is not individually selectable for each channel of Timer F. To make per-channel settings, start by setting this bit to 0 as the setting for Timer F as a unit, and then set noise cancellation mode channel registers 1F (NCMCR1F) and 2F (NCMCR2F).

(5) NCMC — Timer C Noise Cancellation Mode

Selects the noise cancellation mode for timer C. A different operating mode cannot be set for each channel of timer C.

(6) NCMA — Timer A Noise Cancellation Mode

Selects the noise cancellation mode for timer A. A different operating mode cannot be set for each channel of timer A.

Modes of operation are individually selectable for each channel of Timer A. To make per-channel settings, start by setting this bit to 0 as the setting for Timer A as a unit, and then set noise cancellation mode channel register 1A (NCMCR1A) and 2A (NCMCR2A).

Table 21.5 to **Table 21.7** are truth tables for settings that determine the noise cancellation mode.

Table 21.5 Settings that Determine Noise Cancellation Modes for Timer A

Channel Enable	Filter Mode		Channel Register			
TIOR2A. NCEAx	NCMR. NCMA	NCMR. NCMSEL	NCMCR1A. NCM1Ax	NCMCR2A. NCM2Ax	Mode of Cancellation	Unit of Filtering
0	—	—	—	—	Filter invalid	—
1	0	(–)	0	(–)	Premature-transition	Each subblock/ channel
1	0	(–)	1	0	Minimum time-at-level	Each channel
1	0	(–)	1	1	Level accumulation	Each channel
1	1	(0)	—	—	Minimum time-at-level	Each subblock
1	1	1	—	—	Level accumulation	Each subblock

Note 1. x = 0 to 6

Note 2. A noise canceler enable bit Ax (NCEAx) can be set per channel.

Methods of setting noise cancellation for timer A according to the unit of filtering (precondition: channel enable = 1)

- Settings for individual channels:
  After setting the noise cancellation mode bit (NCMA) in the noise cancellation mode register (NCMR) to 0, set even one of the noise cancellation mode channel setting bits (NCM1Ax) in noise cancellation mode channel register 1A (NCMCR1A) of timer A to 1.
- Setting for all channels:
  Set the noise cancellation mode bit (NCMA, the same bit as in “Settings for individual channels” above) in the noise cancellation mode register (NCMR) to 1 or set all of the noise cancellation mode channel setting bits (NCM1Ax) in noise cancellation mode channel register 1A (NCMCR1A) to 0.

Table 21.6 Settings That Determine Noise Cancellation Modes for Timer C

Channel Enable	Filter Mode		Channel Register			
NCCRCx. NCECx	NCMR. NCMC	NCMR. NCMSEL	NCMCR1C. NCM1Cx	NCMCR2C. NCM2Cx	Mode of Cancellation	Unit of Filtering
0	—	—	—	—	Filter invalid	—
1	0	(–)	0	(–)	Premature-transition	Each subblock/ channel
1	0	(–)	1	0	Minimum time-at-level	Each channel
1	0	(–)	1	1	Level accumulation	Each channel
1	1	(0)	—	—	Minimum time-at-level	Each subblock
1	1	1	—	—	Level accumulation	Each subblock

Note 1. x = 0 to 7

Note 2. A noise canceler enable bit Cx (NCECx) can be set for each channel.

Methods of setting noise cancellation for timer C according to the unit of filtering (precondition: channel enable = 1)

- Settings for individual channels:
  After setting the noise cancellation mode bit (NCMC) in the noise cancellation mode register (NCMR) to 0, set even one of the noise cancellation mode channel setting bits (NCM1Cx) in noise cancellation mode channel register 1C (NCMCR1C) of timer C to 1.
- Settings for all channels:
  Set the noise cancellation mode bit (NCMC, the same bit as in “Settings for individual channels” above) in the noise cancellation mode register (NCMR) to 1 or set all of the noise cancellation mode channel setting bits (NCM1Cx) in noise cancellation mode channel register 1C (NCMCR1C) to 0.

Table 21.7 Settings that Determine Noise Cancellation Modes for Timer F

Channel Enable	Filter Mode		Channel Register			
NCCRF. NCEFx	NCMR. NCMF	NCMR. NCMSEL	NCMCR1F. NCM1Fx	NCMCR2F. NCM2Fx	Mode of Cancellation	Unit of Filtering
0	—	—	—	—	Filter invalid	—
1	0	(–)	0	(–)	Premature-transition	Each subblock/ channel
1	0	(–)	1	0	Minimum time-at-level	Each channel
1	0	(–)	1	1	Level accumulation	Each channel
1	1	(0)	—	—	Minimum time-at-level	Each subblock
1	1	1	—	—	Level accumulation	Each subblock

Note 1. x = 0 to 15

Note 2. A noise canceler enable bit Fx (NCEFx) can be set per channel.

Methods of setting noise cancellation for timer F according to the unit of filtering (precondition: channel enable = 1)

- Settings for individual channels: After setting the noise cancellation mode bit (NCMF) in the noise cancellation mode register (NCMR) to 0, set even one of the noise cancellation mode channel setting bits (NCM1Fx) in noise cancellation mode channel register 1F (NCMCR1F) of timer F to 1.
- Setting for all channels: Set the noise cancellation mode bit (NCMF, the same bit as in “Settings for individual channels” above) in the noise cancellation mode register (NCMR) to 1 or set all of the noise cancellation mode channel setting bits (NCM1Fx) in noise cancellation mode channel register 1F (NCMCR1F) to 0.

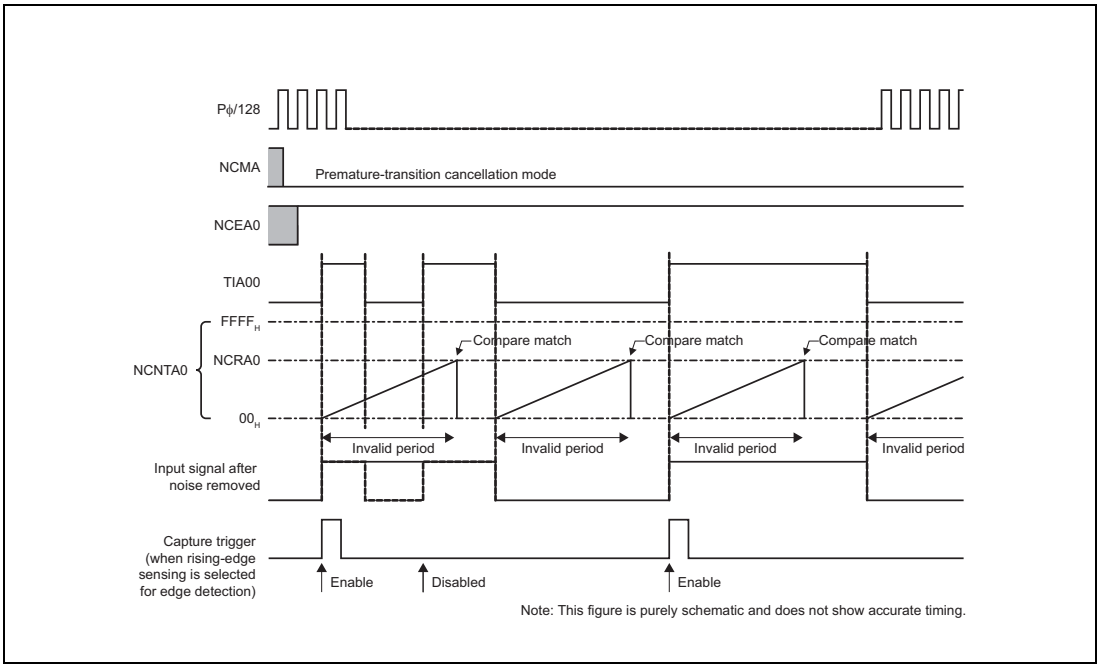


Figure 21.1 Outline of Noise Canceling Operation in Premature-Transition Cancellation Mode

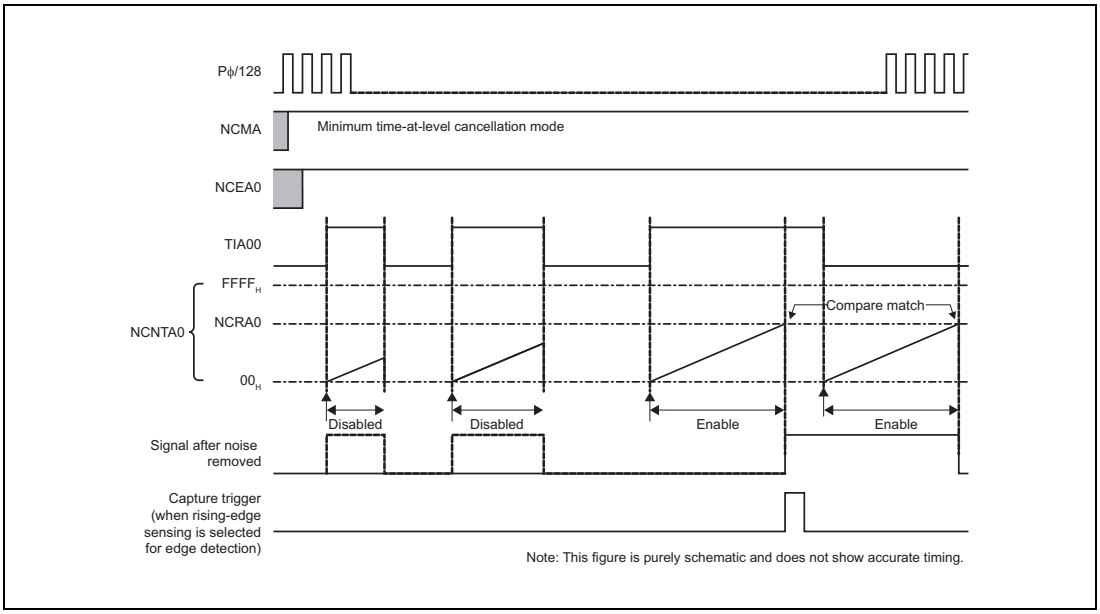


Figure 21.2 Outline of Noise Canceling Operation in Minimum Time-at-Level Cancellation Mode

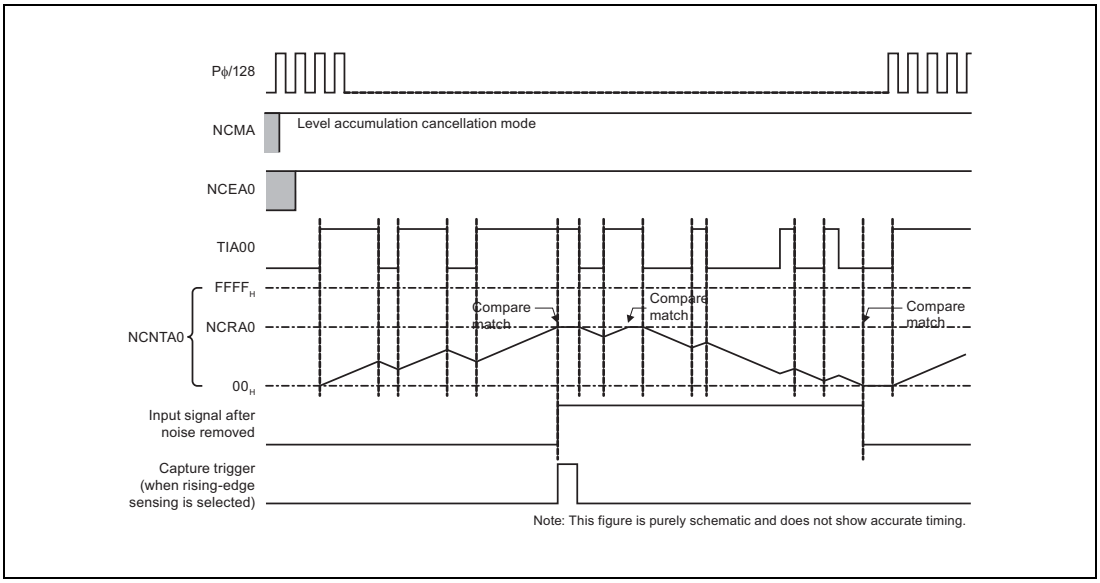


Figure 21.3 Outline of Noise Canceling Operation in Level Accumulation Cancellation Mode

21.3 Prescaler

21.3.1 Operation

ATU-IV includes four general prescalers and a prescaler for the noise-canceler clock.

The general prescalers are implemented as 10-bit down-counters, in which the prescaled clock signals are derived by frequency-dividing the peripheral clock (PCLK) by N ($1 \leq N \leq 1024$). The generated clock signals are supplied to the individual timers via the clock bus. The prescalers for each of the channels operate independently.

The prescaler for the noise-canceler clock is implemented as a 7-bit down-counter. This generates a clock signal by frequency-dividing the peripheral clock (PCLK) by 128. The clock signal thus generated is supplied to the noise cancelers of timers A, C, F, and J.

A clock signal derived by frequency-dividing the peripheral clock frequency by 128 can be selected for the noise canceler clock by the NCCSEL bit in NCMR of the common controller. No other division ratios are available.

The down-counters of the prescalers are initialized to 0000_H by a reset.

21.3.2 Register Description of Prescalers

21.3.2.1 PSCRx — Prescaler Registers x (x = 0 to 3)

Access: 16-bit accessible, but 8-bit inaccessible

Address: FFE6 0080_H (PSCR0)
FFE6 0084_H (PSCR1)
FFE6 0088_H (PSCR2)
FFE6 $008C_H$ (PSCR3)

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	PSCx[9:0]									
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.8 PSCR Register Contents

Bit Position	Bit Name	Function
15 to 10	—	These bits are not used. Fix these bits to 0.
9 to 0	PSCx[9:0]	Division Ratio These bits specify the division ratio for the corresponding prescaler.

Note: x = 0 to 3

Prescaler registers x are 16-bit readable/writable registers, each of which holds a division ratio for one of the four prescalers.

When the value in prescaler register x is changed, the prescaler updates the value on its underflow.

Settable values range from 000_H to $3FF_H$.

The division ratio is obtained from the following expression.

$$\text{Division ratio of prescaler} = \frac{1}{\text{PSCx[9:0]} + 1} \quad \text{(Settable value: 1/1 to 1/1024)}$$

A duty cycle of 50% is not guaranteed for the clock-signal outputs of the prescalers. Instead, the high level is output in one cycle of the PCLK clock and the low level is output in the remaining cycles within the prescaled period.

21.3.3 Details of Operation

21.3.3.1 Starting Prescalers

The prescalers start operating when the PSCE bit in the ATU-IV master enable register is set to 1 and generates a clock with a frequency given by the division ratio in the PSCx bits. While a prescaler is operating, the high level is output for one cycle of the PCLK clock each time the corresponding prescaler counter underflows.

When the setting in the PSCx bits is changed during operation, the division ratio of the output clock is updated on the first subsequent counter underflow.

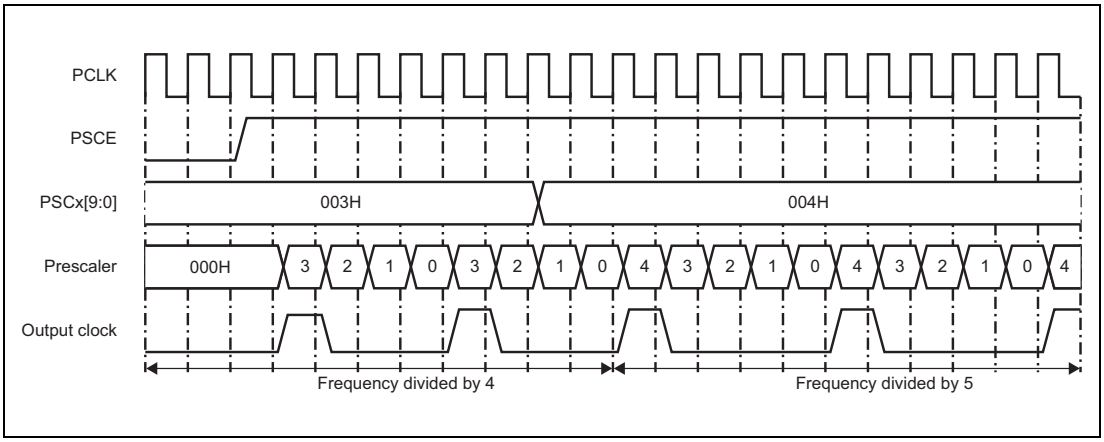


Figure 21.4 Starting Prescaler

21.3.3.2 Stopping and Restarting Operation

The prescaler stops operating when the PSCE bit in the ATU-IV master enable register is cleared to 0. The clock signal stays at the low level while the prescaler is stopped.

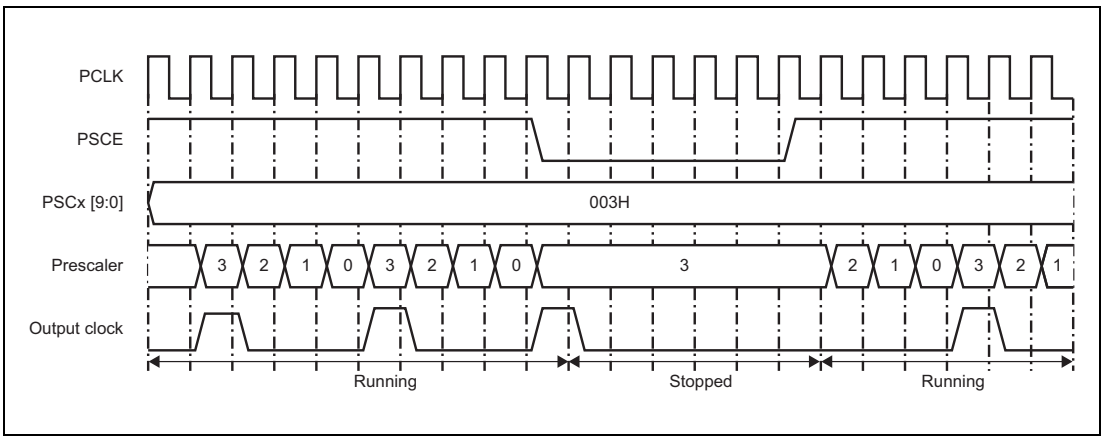


Figure 21.5 Stopping Prescaler

21.4 Timer A

21.4.1 Operation

Timer A includes a 32-bit free-running counter A (TCNTA) and seven 32-bit input capture registers Ax (ICRA0 to ICRA6). TCNTA is a free-running up counter. An interrupt request can be output when the counter overflows.

Seven input capture registers A0 to A6 (ICRA0 to ICRA6) capture the value of free-running counter A (TCNTA) on an assertion of the corresponding external input signal (TIA00 to TIA06). The rising or falling edge, or both edges, can be selected as the trigger for capture. The edge selection is made timer I/O control register 1A (TIOR1A). The interrupt is requested or the A-DMAC is activated at the same timing as capturing.

Noise on the external input signals can be removed by the noise canceler. The pin level of these external signal input pins (TIA00 to TIA06) after noise cancellation can be read and captured. The pin level is captured at the timing of the OCRB6 interrupt request signal from timer B. Input pin (TIA00 to TIA06) signals after noise cancellation can be output as input signals to timer F. Signals on pins TIA00 to TIA06 can be output to timer B, C or D as event signals after noise has been removed and their edges have been extracted. One of TIA00 to 06, or a combination with an input edge are selectable for output to timer B as an event signal (event output 1). Seven event signals TIA00 to 06 are output (event outputs 1B to 1H) to timer B to be used as the capture trigger of the event counter B1(TCNTB1). One of seven event signals TIA00 to 06 or an event signal which is a combination of several input edges is output (event output 2A and 2B) to timer D0 to D9 respectively to be used as the capture trigger of timer offset base registers OSBRD0 to OSBRD 9. The signals of these event output 1 and event output 2A and 2B can be used as the input capture trigger of timer C.

The capture register value can be transferred by the DMAC by activating the DMAC at the capture timing.

Figure 21.6 is a block diagram of timer A.

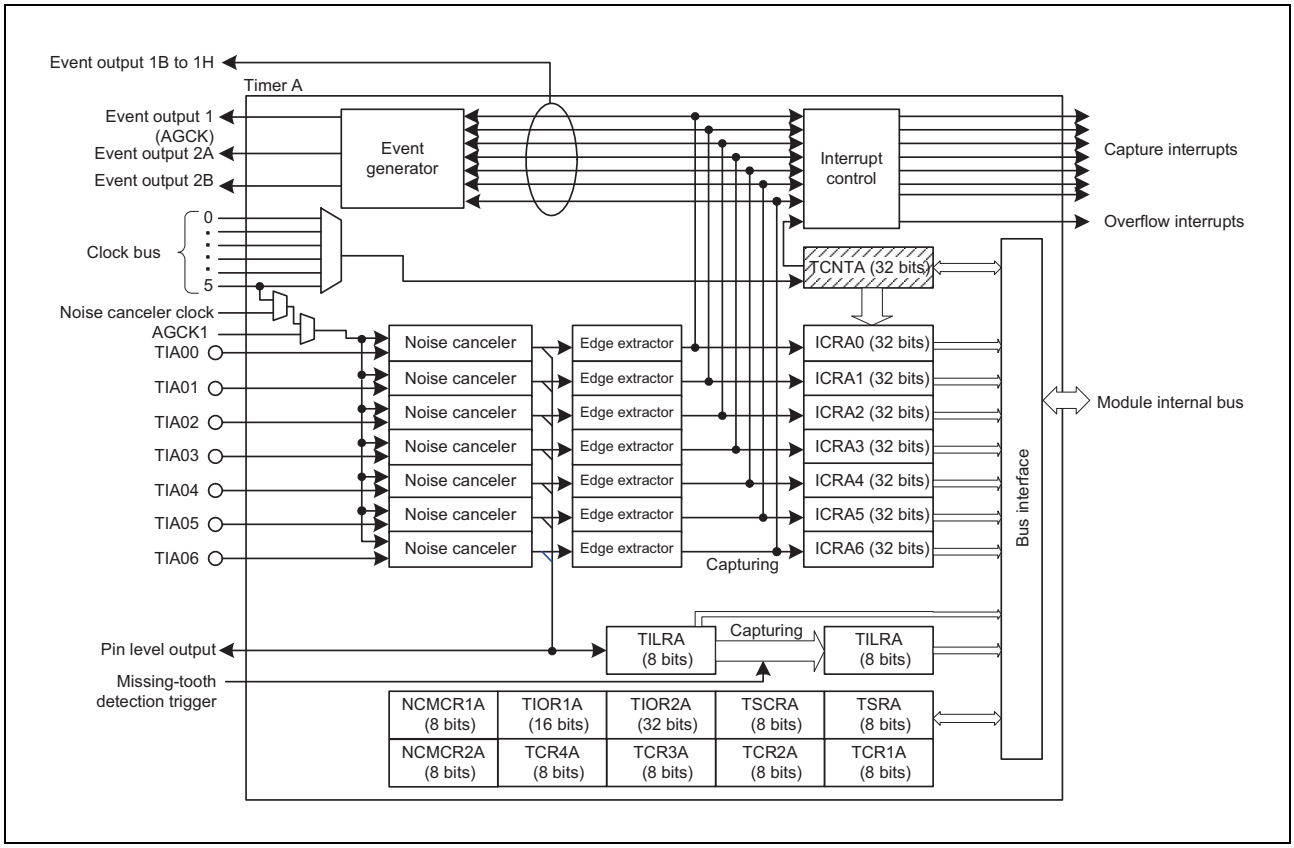


Figure 21.6 Timer A Block Diagram

21.4.2 Timer A Control Registers

21.4.2.1 TCR1A — Timer Control Register 1A

Access: 8-bit accessible

Address: FFE6 0200_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	EVOSEL2A	EVOSEL2B	EVOSEL1			CKSELA		
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.9 TCR1A Register Contents

Bit Position	Bit Name	Function
7	EVOSEL2A	Event Output 2A Select 0: TIA01 is selected as event output 2A 1: TIA02 is selected as event output 2A
6	EVOSEL2B	Event Output 2B Select 0: TIA01 is selected as event output 2B 1: TIA02 is selected as event output 2B
5 to 3	EVOSEL1	Event Output 1 Select 000: No signal is selected as event output 1 001: TIA00 is selected as event output 1 010: TIA01 is selected as event output 1 011: TIA01 and TIA00 input edges are output together. 100: TIA02 is selected as event output 1 101: TIA02 and TIA00 input edges are output together. 110: TIA02 and TIA01 input edges are output together. 111: TIA02, TIA01, and TIA00 input edges are output together.
2 to 0	CKSELA	Clock Select A 000: Clock-bus line 0 (prescaler 0) 001: Clock-bus line 1 (prescaler 1) 010: Clock-bus line 2 (prescaler 2) 011: Clock-bus line 3 (prescaler 3) 100: Clock-bus line 4 (TCLKA) 101: Clock-bus line 5 (TCLKB or multiplied-and-corrected clock) 110: Setting prohibited 111: Setting prohibited

TCR1A is an 8-bit readable/writable register that sets the event output generated from external input signals (TIA00 to TIA02) and the counter clock.

TCR1A is initialized to 00_H by a reset.

(1) EVOSEL2A — Event Output 2A Select

Selects the signal for output on event output 2A as the externally input signal TIA01 or TIA02 (signals after the noise cancellation and edge detection). In timers D0 to D9, the event output 2A signal can be used as a capture trigger for the corresponding timer-offset base registers Dx (OSBRD0 to OSBRD9).

Timer C can use this event output 2A signal as a capture trigger.

(2) EVOSEL2B — Event Output 2B Select

Selects the signal for output on event output 2B as the externally input signal TIA01 or TIA02 (signals after the noise cancellation and edge detection). In timer D0 to D9, the event output 2B signal can be used as a capture trigger for the corresponding timer-offset base register Dx (OSBRD0 to OSBRD9).

Timer C can use this event output 2B signal as a capture trigger.

(3) EVOSEL1 — Event Output 1 Select

These bits select the signal for output on event output 1 as one of the externally input signal TIA00 to TIA02 (signals after the noise cancellation and edge detection). In timer B, the event output 1 signal can be used as an event output 1 to timer B and timer C. Furthermore, input edge combinations of TIA00 and TIA01, TIA00 and TIA02, TIA01 and TIA02, and TIA00, TIA01, and TIA02 can be output together as an event output 1 to timer B and timer C.

(4) CKSELA — Clock Select A

These bits select the signal on one of clock-bus lines 0 to 5 as the clock signal for counting. The signal on lines 0 to 3 have been frequency-divided by prescalers 0 to 3, respectively. Clock-bus line 4 supplies externally input clock A (TCLKA). Clock-bus line 5 supplies externally input clock B (TCLKB) or the multiplied-and-corrected clock output by timer B.

Stop timer A before making or changing the counter-clock selection.

CAUTION

The edge of an external input clock is detected before it is output on a clock bus. When using external input clock A or B, select the edge to be detected by setting the edge select bit in the clock bus control register (CBCNT).

21.4.2.2 TCR2A — Timer Control Register 2A

Access: 8-bit accessible

Address: FFE6 0202_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	EVOSELE1E	EVOSELE1						
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.10 TCR2A Register Contents

Bit Position	Bit Name	Function
7	EVOSELE1E	Event Output 1 Extension Select Enable 0: Event output 1 extension selection is disabled. 1: Event output 1 extension selection is enabled.
6 to 0	EVOSELE1	Event Output 1 Extension Select These bits select the combination of inputs (TIA00 to 06) to be output as an event. b6: 0: TIA06 input edge is not output 1: TIA06 input edge is output b5: 0: TIA05 input edge is not output 1: TIA05 input edge is output b4: 0: TIA04 input edge is not output 1: TIA04 input edge is output b3: 0: TIA03 input edge is not output 1: TIA03 input edge is output b2: 0: TIA02 input edge is not output 1: TIA02 input edge is output b1: 0: TIA01 input edge is not output 1: TIA01 input edge is output b0: 0: TIA00 input edge is not output 1: TIA00 input edge is output

(1) EVOSELE1E — Event Output 1 Extension Select Enable

Event Output 1 Extension Select Enable (EVOSELE1E) is an 8-bit readable/writable register to enable or disable the event output 1 extension selection.

EVOSELE1E is initialized to 00_H by a reset.

(2) EVOSELE1 — Event Output 1 Extension Select Register

Event Output 1 Extension Select register (EVOSELE1) is an 8-bit readable/writable register to set event output 1 of external input edges TIA00 to TIA06 when EVOSELE1E is enabled.

EVOSELE1 is initialized to 00_H by a reset.

Any one of external input signals TIA00 to TIA06 (signals after noise cancellation and edge detection) or combination of these input edges can be output to timer B or timer C as event 1 by the settings of this register.

When this function is enabled, event output selection specified by TCR1A is disabled.

21.4.2.3 TCR3A — Timer Control Register 3A

Access: 8-bit accessible

Address: FFE6 0204_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	EVOSELE2AE	EVOSELE2A						
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.11 TCR3A Register Contents

Bit Position	Bit Name	Function
7	EVOSELE2AE	Event Output 2A Extension Enable 0: Event output 2A extension selection is disabled. 1: Event output 2A extension selection is enabled.
6 to 0	EVOSELE2A	Event Output 2A Extension Select These bits select the combination of inputs (TIA00 to 06) to be output as an event. b6: 0: TIA06 input edge is not output 1: TIA06 input edge is output b5: 0: TIA05 input edge is not output 1: TIA05 input edge is output b4: 0: TIA04 input edge is not output 1: TIA04 input edge is output b3: 0: TIA03 input edge is not output 1: TIA03 input edge is output b2: 0: TIA02 input edge is not output 1: TIA02 input edge is output b1: 0: TIA01 input edge is not output 1: TIA01 input edge is output b0: 0: TIA00 input edge is not output 1: TIA00 input edge is output

(1) EVOSELE2AE — Event Output 2A Extension Select Enable

Event Output 2A Extension Select Enable (EVOSELE2AE) is an 8-bit readable/writable register to enable or disable the event output 2A extension selection. EVOSELE2AE is initialized to 00_H by a reset.

(2) EVOSELE2A — Event Output 2A Extension Select Register

Event Output 2A Extension Select register (EVOSELE2A) is an 8-bit readable/writable register to set event output 2A of external input edges TIA00 to TIA06 when EVOSELE2AE is enabled.

EVOSELE2A is initialized to 00_H by a reset.

Any one of external input signals TIA00 to TIA06 (signals after noise cancellation and edge detection) or combination of these input edges can be output as event 2A by the settings of this register. Timer D0 to timer D9 can use this event 2A signal as the capture trigger of timer offset base registers Dx (OSBRD0 to OSBRD9).

Timer C can use this event output 2A signal as a capture trigger.

When this function is enabled, event output selection specified by TCR1A is disabled.

21.4.2.4 TCR4A — Timer Control Register 4A

Access: 8-bit accessible

Address: FFE6 0206_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	EVOSELE2BE	EVOSELE2B						
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.12 TCR4A Register Contents

Bit Position	Bit Name	Function
7	EVOSELE2BE	Event Output 2B Extension Enable 0: Event output 2B extension selection is disabled. 1: Event output 2B extension selection is enabled.
6 to 0	EVOSELE2B	Event Output 2B Extension Select These bits select the combination of inputs (TIA00 to 06) to be output as an event. b6: 0: TIA06 input edge is not output 1: TIA06 input edge is output b5: 0: TIA05 input edge is not output 1: TIA05 input edge is output b4: 0: TIA04 input edge is not output 1: TIA04 input edge is output b3: 0: TIA03 input edge is not output 1: TIA03 input edge is output b2: 0: TIA02 input edge is not output 1: TIA02 input edge is output b1: 0: TIA01 input edge is not output 1: TIA01 input edge is output b0: 0: TIA00 input edge is not output 1: TIA00 input edge is output

(1) EVOSELE2BE — Event Output 2B Extension Select Enable

Event Output 2B Extension Select Enable (EVOSELE2BE) is an 8-bit readable/writable register to enable or disable the event output 2B extension selection.

EVOSELE2BE is initialized to 00_H by a reset.

(2) EVOSELE2B — Event Output 2B Extension Select Register

Event Output 2B Extension Select register (EVOSELE2B) is an 8-bit readable/writable register to set event output 2B of external input edges TIA00 to TIA06 when EVOSELE2BE is enabled.

EVOSELE2B is initialized to 00_H by a reset.

Any one of external input signals TIA00 to TIA06 (signals after noise cancellation and edge detection) or a combination of them can be output as event 2B with input edge by the settings of this register. Timer D0 to timer D9 can use this event 2B signal as the capture trigger of timer-offset base registers Dx (OSBRD0 to OSBRD9).

Timer C can use this event output 2B signal as a capture trigger.

When this function is enabled, event output selection specified by TCR1A is disabled.

21.4.2.5 NCMCR1A — Noise Cancellation Mode Channel Register 1A

Access: 8-bit accessible

Address: FFE6 020C$_H$

Value after reset: 00$_H$

Bit	7	6	5	4	3	2	1	0
	—	NCM1A6	NCM1A5	NCM1A4	NCM1A3	NCM1A2	NCM1A1	NCM1A0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.13 NCMCR1A Register Contents

Bit Position	Bit Name	Function
7	—	This bit is not used. Fix this bit to 0.
6	NCM1A6	Noise Cancellation Mode of Channel 6 Specify the operating mode of noise canceler in channel 6. 0: Premature-transition cancellation mode 1: Minimum time-at-level cancellation mode (When NCMA = 0 and NCM2A6 = 0) 1: Level accumulation cancellation mode (When NCMA = 0 and NCM2A6 = 1)
5	NCM1A5	Noise Cancellation Mode of Channel 5 Specify the operation mode of noise canceler in channel 5. 0: Premature-transition cancellation mode 1: Minimum time-at-level cancellation mode (When NCMA = 0 and NCM2A5 = 0) 1: Level accumulation cancellation mode (When NCMA = 0 and NCM2A5 = 1)
4	NCM1A4	Noise Cancellation Mode of Channel 4 Specify the operation mode of noise canceler in channel 4. 0: Premature-transition cancellation mode 1: Minimum time-at-level cancellation mode (When NCMA = 0 and NCM2A4 = 0) 1: Level accumulation cancellation mode (When NCMA = 0 and NCM2A4 = 1)
3	NCM1A3	Noise Cancellation Mode of Channel 3 Specify the operation mode of noise canceler in channel 3. 0: Premature-transition cancellation mode 1: Minimum time-at-level cancellation mode (When NCMA = 0 and NCM2A3 = 0) 1: Level accumulation cancellation mode (When NCMA = 0 and NCM2A3 = 1)
2	NCM1A2	Noise Cancellation Mode of Channel 2 Specify the operation mode of noise canceler in channel 2. 0: Premature-transition cancellation mode 1: Minimum time-at-level cancellation mode (When NCMA = 0 and NCM2A2 = 0) 1: Level accumulation cancellation mode (When NCMA = 0 and NCM2A2 = 1)
1	NCM1A1	Noise Cancellation Mode of Channel 1 Specify the operation mode of noise canceler in channel 1. 0: Premature-transition cancellation mode 1: Minimum time-at-level cancellation mode (When NCMA = 0 and NCM2A1 = 0) 1: Level accumulation cancellation mode (When NCMA = 0 and NCM2A1 = 1)
0	NCM1A0	Noise Cancellation Mode of Channel 0 Specify the operation mode of noise canceler in channel 0. 0: Premature-transition cancellation mode 1: Minimum time-at-level cancellation mode (When NCMA = 0 and NCM2A0 = 0) 1: Level accumulation cancellation mode (When NCMA = 0 and NCM2A0 = 1)

NCMCR1A is an 8-bit readable/writable register that selects the operation mode of noise canceler for each channel unit.

In premature-transition cancellation mode, after an input signal level change is detected, any input signal level change in the specified period is ignored. This mode regards a signal level change in the specified period as noise after the first level change is detected.

In level accumulation cancellation mode, input signal levels are accumulated and the input level is regarded to have reached 0 or 1 when the accumulation result becomes 0 or the specified value.

Each period is specified by the noise canceler register of each channel and the noise canceler counter measures time.

Outline of noise canceling operation (example of TIA00 input signal of timer A) is shown in **Figure 21.1** (in premature-transition cancellation mode), **Figure 21.2** (in minimum time-at-level cancellation mode), and **Figure 21.3** (in level accumulation cancellation mode).

Each channel detects the edge of a signal that has passed through the noise canceler. **Figure 21.1**, **Figure 21.2**, and **Figure 21.3** show an example of detecting the rising edge of a signal that has passed through the noise canceler.

(1) NCM1A6 — Noise Cancellation Mode of Channel 6

Specify the operating mode of noise canceler in channel 6.

(2) NCM1A5 — Noise Cancellation Mode of Channel 5

Specify the operation mode of noise canceler in channel 5.

(3) NCM1A4 — Noise Cancellation Mode of Channel 4

Specify the operation mode of noise canceler in channel 4.

(4) NCM1A3 — Noise Cancellation Mode of Channel 3

Specify the operation mode of noise canceler in channel 3.

(5) NCM1A2 — Noise Cancellation Mode of Channel 2

Specify the operation mode of noise canceler in channel 2.

(6) NCM1A1 — Noise Cancellation Mode of Channel 1

Specify the operation mode of noise canceler in channel 1.

(7) NCM1A0 — Noise Cancellation Mode of Channel 0

Specify the operation mode of noise canceler in channel 0.

CAUTIONS

1. **This register is only effective when the NCMA bit of the noise cancellation mode register (NCMR) in the common control unit is 0. Furthermore, when the corresponding channel bit (NCM1Ax) in this register is set to 1, the mode can be set to premature-transition cancellation mode, minimum time-at-level cancellation mode, or level accumulation cancellation mode according to the state of the corresponding channel bit in the noise cancellation mode channel register 2A (NCMCR2A).**
2. **x is 0 to 6.**

Table 21.14 shows the truth table for setting the noise cancellation mode.

Table 21.14 Truth Table for Noise Cancellation Modes of Timer A

Channel Enable	Filter Mode		Channel Register			
TIOR2A. NCEAx	NCMR. NCMA	NCMR. NCMSEL	NCMCR1A. NCM1Ax	NCMCR2A. NCM2Ax	Mode of Cancellation	Unit of Filtering
0	—	—	—	—	Filter invalid	—
1	0	(—)	0	(—)	Premature-transition	Each subblock/ channel
1	0	(—)	1	0	Minimum time-at-level	Each channel
1	0	(—)	1	1	Level accumulation	Each channel
1	1	(0)	—	—	Minimum time-at-level	Each subblock
1	1	1	—	—	Level accumulation	Each subblock

Note 1. x = 0 to 6

Note 2. An enable bit (NCEAx) can be set per channel.

Methods of setting noise cancellation for timer A according to the unit of filtering (precondition: channel enable = 1)

- Settings for individual channels:
  After setting the noise cancellation mode bit (NCMA) in the noise cancellation mode register (NCMR) to 0, set even one of the noise cancellation mode channel setting bits (NCM1Ax) in noise cancellation mode channel register 1A (NCMCR1A) of timer A to 1.
- Setting for all channels:
  Set the noise cancellation mode bit (NCMA, the same bit as in “Settings for individual channels” above) in the noise cancellation mode register (NCMR) to 1 or set all of the noise cancellation mode channel setting bits (NCM1Ax) in noise cancellation mode channel register 1A (NCMCR1A) to 0.

21.4.2.6 NCMCR2A — Noise Cancellation Mode Channel Register 2A

Access: 8-bit accessible

Address: FFE6 020E$_H$

Value after reset: 00$_H$

Bit	7	6	5	4	3	2	1	0
	—	NCM2A6	NCM2A5	NCM2A4	NCM2A3	NCM2A2	NCM2A1	NCM2A0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.15 NCMCR2A Register Contents

Bit Position	Bit Name	Function
7	—	This bit is not used. Fix this bit to 0.
6	NCM2A6	Noise Cancellation Mode 2 of Channel 6 Specify the operating mode of noise canceler in channel 6. 0: Minimum time-at-level cancellation mode (When NCMA = 0 and NCM1A6 = 1) 1: Level accumulation cancellation mode (When NCMA = 0 and NCM1A6 = 1)
5	NCM2A5	Noise Cancellation Mode 2 of Channel 5 Specify the operation mode of noise canceler in channel 5. 0: Minimum time-at-level cancellation mode (When NCMA = 0 and NCM1A5 = 1) 1: Level accumulation cancellation mode (When NCMA = 0 and NCM1A5 = 1)
4	NCM2A4	Noise Cancellation Mode 2 of Channel 4 Specify the operation mode of noise canceler in channel 4. 0: Minimum time-at-level cancellation mode (When NCMA = 0 and NCM1A4 = 1) 1: Level accumulation cancellation mode (When NCMA = 0 and NCM1A4 = 1)
3	NCM2A3	Noise Cancellation Mode 2 of Channel 3 Specify the operation mode of noise canceler in channel 3. 0: Minimum time-at-level cancellation mode (When NCMA = 0 and NCM1A3 = 1) 1: Level accumulation cancellation mode (When NCMA = 0 and NCM1A3 = 1)
2	NCM2A2	Noise Cancellation Mode 2 of Channel 2 Specify the operation mode of noise canceler in channel 2. 0: Minimum time-at-level cancellation mode (When NCMA = 0 and NCM1A2 = 1) 1: Level accumulation cancellation mode (When NCMA = 0 and NCM1A2 = 1)
1	NCM2A1	Noise Cancellation Mode 2 of Channel 1 Specify the operation mode of noise canceler in channel 1. 0: Minimum time-at-level cancellation mode (When NCMA = 0 and NCM1A1 = 1) 1: Level accumulation cancellation mode (When NCMA = 0 and NCM1A1 = 1)
0	NCM2A0	Noise Cancellation Mode 2 of Channel 0 Specify the operation mode of noise canceler in channel 0. 0: Minimum time-at-level cancellation mode (When NCMA = 0 and NCM1A0 = 1) 1: Level accumulation cancellation mode (When NCMA = 0 and NCM1A0 = 1)

NCMCR2A is an 8-bit readable/writable register that selects the operation mode of noise canceler for each channel unit.

In level accumulation cancellation mode, input signal levels are accumulated and the input level is regarded to have reached 0 or 1 when the accumulation result becomes 0 or the specified value.

Each period is specified by the noise canceler register of each channel and the noise canceler counter measures time.

Outline of noise canceling operation (example of TIA00 input signal of timer A) is shown in **Figure 21.1** (in premature-transition cancellation mode) and **Figure 21.3** (in level accumulation cancellation mode).

Each channel detects the edge of a signal that has passed through the noise canceler. **Figure 21.1** and **Figure 21.3** show an example of detecting the rising edge of a signal that has passed through the noise canceler.

(1) NCM2A6 — Noise Cancellation Mode 2 of Channel 6

Specify the operating mode of noise canceler in channel 6.

(2) NCM2A5 — Noise Cancellation Mode 2 of Channel 5

Specify the operation mode of noise canceler in channel 5.

(3) NCM2A4 — Noise Cancellation Mode 2 of Channel 4

Specify the operation mode of noise canceler in channel 4.

(4) NCM2A3 — Noise Cancellation Mode 2 of Channel 3

Specify the operation mode of noise canceler in channel 3.

(5) NCM2A2 — Noise Cancellation Mode 2 of Channel 2

Specify the operation mode of noise canceler in channel 2.

(6) NCM2A1 — Noise Cancellation Mode 2 of Channel 1

Specify the operation mode of noise canceler in channel 1.

(7) NCM2A0 — Noise Cancellation Mode 2 of Channel 0

Specify the operation mode of noise canceler in channel 0.

CAUTIONS

1. **This register is only effective when the NCMA bit in the noise cancellation mode register (NCMR) in the common control unit is 0 and the corresponding channel bit (NCM1Ax) in noise cancellation mode channel register 1A (NCMCR1A) of timer A is set to 1.**
2. **x = 0 to 6**

Table 21.16 shows the truth table for setting the noise cancellation mode.

Table 21.16 Truth Table for Noise Cancellation Modes of Timer A

Channel Enable	Filter Mode		Channel Register			
TIOR2A. NCEAx	NCMR. NCMA	NCMR. NCMSEL	NCMCR1A. NCM1Ax	NCMCR2A. NCM2Ax	Mode of Cancellation	Unit of Filtering
0	—	—	—	—	Filter invalid	—
1	0	(—)	0	(—)	Premature-transition	Each subblock/ channel
1	0	(—)	1	0	Minimum time-at-level	Each channel
1	0	(—)	1	1	Level accumulation	Each channel
1	1	(0)	—	—	Minimum time-at-level	Each subblock
1	1	1	—	—	Level accumulation	Each subblock

Note 1. x = 0 to 6

Note 2. An enable bit (NCEAx) can be set per channel.

Methods of setting noise cancellation for timer A according to the unit of filtering (precondition: channel enable = 1)

- Settings for individual channels:
  After setting the noise cancellation mode bit (NCMA) in the noise cancellation mode register (NCMR) to 0, set even one of the noise cancellation mode channel setting bits (NCM1Ax) in noise cancellation mode channel register 1A (NCMCR1A) of timer A to 1.
- Setting for all channels:
  Set the noise cancellation mode bit (NCMA, the same bit as in “Settings for individual channels” above) in the noise cancellation mode register (NCMR) to 1 or set all of the noise cancellation mode channel setting bits (NCM1Ax) in noise cancellation mode channel register 1A (NCMCR1A) to 0.

21.4.2.7 TIOR1A — Timer I/O Control Register 1A

Access: 8-bit and 16-bit accessible

Address: FFE6 0210_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	IOA6		IOA5		IOA4		IOA3		IOA2		IOA1		IOA0	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.17 TIOR1A Register Contents

Bit Position	Bit Name	Function
15, 14	—	These bits are not used. Fix these bits to 0.
13, 12	IOA6	I/O Control A6 00: TIA06 input capturing is prohibited. 01: TCNTA is captured in ICRA6 on the rising edge of TIA06. 10: TCNTA is captured in ICRA6 on the falling edge of TIA06. 11: TCNTA is captured in ICRA6 on both edges of TIA06.
11, 10	IOA5	I/O Control A5 00: TIA05 input capturing is prohibited. 01: TCNTA is captured in ICRA5 on the rising edge of TIA05 10: TCNTA is captured in ICRA5 on the falling edge of TIA05 11: TCNTA is captured in ICRA5 on both edges of TIA05
9, 8	IOA4	I/O Control A4 00: TIA04 input capturing is prohibited. 01: TCNTA is captured in ICRA4 on the rising edge of TIA04 10: TCNTA is captured in ICRA4 on the falling edge of TIA04 11: TCNTA is captured in ICRA4 on both edges of TIA04
7, 6	IOA3	I/O Control A3 00: TIA03 input capturing is prohibited. 01: TCNTA is captured in ICRA3 on the rising edge of TIA03 10: TCNTA is captured in ICRA3 on the falling edge of TIA03 11: TCNTA is captured in ICRA3 on both edges of TIA03
5, 4	IOA2	I/O Control A2 00: TIA02 input capturing is prohibited. 01: TCNTA is captured in ICRA2 on the rising edge of TIA02 10: TCNTA is captured in ICRA2 on the falling edge of TIA02 11: TCNTA is captured in ICRA2 on both edges of TIA02
3, 2	IOA1	I/O Control A1 00: TIA01 input capturing is prohibited. 01: TCNTA is captured in ICRA1 on the rising edge of TIA01 10: TCNTA is captured in ICRA1 on the falling edge of TIA01 11: TCNTA is captured in ICRA1 on both edges of TIA01
1, 0	IOA0	I/O Control A0 00: TIA00 input capturing is prohibited. 01: TCNTA is captured in ICRA0 on the rising edge of TIA00 10: TCNTA is captured in ICRA0 on the falling edge of TIA00 11: TCNTA is captured in ICRA0on both edges of TIA00

Timer I/O Control Register 1A (TIOR1A) is a 16-bit readable/writable register that sets the edge of external inputs (TIA00 to TIA06) to be detected.

TIOR1A is initialized to 0000_H by a reset.

(1) IOA0 to 6 — I/O Control A0 to 6

These bits select the edge of external inputs (TIA00 to TIA06) that is to be extracted for use in input-capture triggering. When these bits are set to 00_B, input capturing is not performed. When the bit is set to 01, 10 or 11, the contents of free-running counter A0 to A5 (TCNTA) are transferred to input capture register A0 to A6 (ICRA0 to ICRA6) on detection of the selected edge from one of the external inputs.

Edge extraction is synchronized with the PCLK clock. Make sure that the frequency of the Pφ clock is at least twice the frequency of the external input signal. Otherwise, edge extraction will not be performed correctly.

Edge extraction is performed for signals that have passed through the noise canceler. When the noise canceler is disabled, the selected edge is simply extracted from the external inputs (TIA00 to TIA06). When the noise canceler is enabled, the selected edge is extracted from the signals after noise removal.

The edge detection signals of TIA00 to TIA06 are directly output to event output 1B to 1H. TIA00 corresponds to event output 1B, TIA01 to event output 1C, TIA02 to event output 1D, TIA03 to event output 1E, TIA04 to event output 1F, TIA05 to event output 1G, and TIA06 to event output 1H respectively.

Furthermore, the edge extracted here is output to other timer modules by setting the event output 1 and event output 2 select bits (EVOSEL1, EVOSEL2) in TCR1A, the event output 1 extension select bit in TCR2A, the event output 2A extension select bit in TCR3A, and the event output 2B extension select bit in TCR4A. (Signals to be output after edge extraction are generated in positive logic.)

21.4.2.8 TIOR2A — Timer I/O Control Register 2A

Access: 8-bit/16-bit/32-bit accessible

Address: FFE6 0214_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	NCKGA6	NCKGA5	NCKGA4	NCKGA3	NCKGA2	NCKGA1	NCKGA0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	NCKA6	NCKA5	NCKA4	NCKA3	NCKA2	NCKA1	NCKA0	—	NCEA6	NCEA5	NCEA4	NCEA3	NCEA2	NCEA1	NCEA0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.18 TIOR2A Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 23	—	These bits are not used. Fix these bits to 0.
22	NCKGA6	Noise Canceler Clock Select GA6 0: The clock selected by NCKA6 is selected as the count source clock of NCNTA6. 1: Noise canceler count clock (AGCK1) is selected as the count source clock of NCNTA6.
21	NCKGA5	Noise Canceler Clock Select GA5 0: The clock selected by NCKA5 is selected as the count source clock of NCNTA5. 1: Noise canceler count clock (AGCK1) is selected as the count source clock of NCNTA5.
20	NCKGA4	Noise Canceler Clock Select GA4 0: The clock selected by NCKA4 is selected as the count source clock of NCNTA4. 1: Noise canceler count clock (AGCK1) is selected as the count source clock of NCNTA4.
19	NCKGA3	Noise Canceler Clock Select GA3 0: The clock selected by NCKA3 is selected as the count source clock of NCNTA3. 1: Noise canceler count clock (AGCK1) is selected as the count source clock of NCNTA3.
18	NCKGA2	Noise Canceler Clock Select GA2 0: The clock selected by NCKA2 is selected as the count source clock of NCNTA2. 1: Noise canceler count clock (AGCK1) is selected as the count source clock of NCNTA2.
17	NCKGA1	Noise Canceler Clock Select GA1 0: The clock selected by NCKA1 is selected as the count source clock of NCNTA1. 1: Noise canceler count clock (AGCK1) is selected as the count source clock of NCNTA1.
16	NCKGA0	Noise Canceler Clock Select GA0 0: The clock selected by NCKA0 is selected as the count source clock of NCNTA0. 1: Noise canceler count clock (AGCK1) is selected as the count source clock of NCNTA0.
15	—	This bit is not used. Fix this bit to 0.

Table 21.18 TIOR2A Register Contents (2/2)

Bit Position	Bit Name	Function
14	NCKA6	Noise Canceler Clock Select A6 0: Noise canceler count clock (Pϕ or Pϕ/128) is selected as the count source clock of NCNTA6. 1: Clock-bus line 5 is selected as the count source clock of NCNTA6.
13	NCKA5	Noise Canceler Clock Select A5 0: Noise canceler count clock (Pϕ or Pϕ/128) is selected as the signal for counting by NCNTA5 1: Clock-bus line 5 is selected as the signal for counting by NCNTA5
12	NCKA4	Noise Canceler Clock Select A4 0: Noise canceler count clock (Pϕ or Pϕ/128) is selected as the signal for counting by NCNTA4 1: Clock-bus line 5 is selected as the signal for counting by NCNTA4
11	NCKA3	Noise Canceler Clock Select A3 0: Noise canceler count clock (Pϕ or Pϕ/128) is selected as the signal for counting by NCNTA3 1: Clock-bus line 5 is selected as the signal for counting by NCNTA3
10	NCKA2	Noise Canceler Clock Select A2 0: Noise canceler count clock (Pϕ or Pϕ/128) is selected as the signal for counting by NCNTA2 1: Clock-bus line 5 is selected as the signal for counting by NCNTA2
9	NCKA1	Noise Canceler Clock Select A1 0: Noise canceler count clock (Pϕ or Pϕ/128) is selected as the signal for counting by NCNTA1 1: Clock-bus line 5 is selected as the signal for counting by NCNTA1
8	NCKA0	Noise Canceler Clock Select A0 0: Noise canceler count clock (Pϕ or Pϕ/128) is selected as the signal for counting by NCNTA0 1: Clock-bus line 5 is selected as the signal for counting by NCNTA0
7	—	This bit is not used. Fix this bit to 0.
6	NCEA6	Noise Canceler Enable A6 0: Noise cancelers for TIA06 are disabled. 1: Noise cancelers for TIA06 are enabled.
5	NCEA5	Noise Canceler Enable A5 0: Noise cancelers for TIA05 are disabled 1: Noise cancelers for TIA05 are enabled
4	NCEA4	Noise Canceler Enable A4 0: Noise cancelers for TIA04 are disabled 1: Noise cancelers for TIA04 are enabled
3	NCEA3	Noise Canceler Enable A3 0: Noise cancelers for TIA03 are disabled 1: Noise cancelers for TIA03 are enabled
2	NCEA2	Noise Canceler Enable A2 0: Noise cancelers for TIA02 are disabled 1: Noise cancelers for TIA02 are enabled
1	NCEA1	Noise Canceler Enable A1 0: Noise cancelers for TIA01 are disabled 1: Noise cancelers for TIA01 are enabled
0	NCEA0	Noise Canceler Enable A0 0: Noise cancelers for TIA00 are disabled 1: Noise cancelers for TIA00 are enabled

Timer I/O Control Register 2A (TIOR2A) is a 32-bit readable/writable register that selects the noise canceler clock and enables and disables the noise cancelers for externally input signals (TIA00 to TIA06).

TIOR2A is initialized to 0000 0000_H by a reset.

(1) NCKGA0 to 6 — Noise Canceler Clock Select G A0 to 6

These bits select the count source clock of noise canceler counters A0 to 6 (NCNTGA0 to 6).

The noise canceler count clock selected by the noise canceler clock select bits (NCKA0 to 6) or AGCK1 can be selected as the count source clock.

(2) NCKA0 to 6 — Noise Canceler Clock Select A0 to 6

These bits select the count source clock of noise canceler counter A0 to 6 (NCNTA0 to 6).

The noise canceler count clock or the signal on clock-bus line 5 can be selected as the signal for counting. Either the Pϕ clock frequency divided by 128 or the Pϕ clock can be selected as the noise canceler count clock by setting the NCCSEL bit of the common controller.

(3) NCEA0 to 6 — Noise Canceler Enable A0 to 6

These bits enable and disable the noise cancelers for externally input signals (TIA00 to TIA06).

When a level change on externally input signals TIA00 to TIA06 is detected while this bit is set to 1, it is processed in premature-transition cancellation mode, minimum time-at-level cancellation mode, or level accumulation cancellation mode depending on the setting in the noise cancellation mode register (NCMR) of the common controller, noise cancellation mode channel register 1A (NCMCR1A) of timer A, or noise cancellation mode channel register 2A (NCMCR2A) of timer A.

In premature-transition cancellation mode, when a level change of the externally input signal is detected, the change is output as the signal whose noise is removed and the corresponding noise canceler counter (NCNTA0 to NCNTA6) is started for counting up. Subsequent level changes are masked until the value in the counter reaches the value in the noise canceler register (NCRA0 to NCRA6). The level of the externally input signal is output as the signal following noise cancellation at the time of the match in comparison.

When these bits are cleared while the counter (NCNTA0 to NCNTA6) is being incremented, counting continues until the values in the counter and the noise canceler register match. The subsequent level changes are also masked over this period.

In minimum time-at-level cancellation mode, when a level change of the externally input signal is detected, the corresponding noise canceler counter (NCNTA0 to NCNTA6) is started for counting up. If subsequent level changes are not detected until the value in the counter reaches the value in the noise canceler register (NCRA0 to NCRA6), the previously accepted level change is output as the signal following noise cancellation at the time of the match in comparison. When the subsequent level change is detected before the values in the counter and noise canceler register match, all the changes are treated as noise. Therefore the signal whose noise is removed is not changed.

When these bits are cleared while the counter (NCNTA0 to NCNTA6) is being incremented, counting continues until the values in the counter and the noise canceler register match or a level change on the externally input signal is detected.

In level accumulation cancellation mode, the corresponding noise canceler counter A0 to A6 (NCNTA0 to NCNTA6) increments or decrements according to the input signal level. It increments at the input high level and decrements at the input low level. Up-counting continues until the noise canceler counter value matches the noise canceler register A0 to A6 (NCRA0 to NCRA6) value. Down-counting continues until the noise canceler counter becomes 00_H.

If the counter is counting up, the noise canceler output is updated to 1 in response to a match in comparison (NCRA0 to NCRA6). If the counter is counting down, the noise canceler output is updated to 0 in response to a match in comparison (00_H).

In minimum time-at-level cancellation mode and premature-transition cancellation mode, level changes are always detected by Pϕ regardless of the selected noise-canceler clock. In level accumulation cancellation mode, input level sampling is performed by the clock selected by the noise canceler clock select bits A0 to A6 (NCKA0 to NCKA6).

For examples of operation in each cancellation mode, see **Figure 21.1**, **Figure 21.2**, and **Figure 21.3**.

21.4.2.9 TSRA — Timer Status Register A

Access: 8-bit accessible

Address: FFE6 0208_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	OVFA	ICFA6	ICFA5	ICFA4	ICFA3	ICFA2	ICFA1	ICFA0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 21.19 TSRA Register Contents

Bit Position	Bit Name	Function
7	OVFA	Overflow Flag A 0: Indicates that TCNTA has not overflowed 1: Indicates that TCNTA has overflowed
6	ICFA6	Input Capture Flag A6 0: No input capture has occurred 1: Input capture has occurred
5	ICFA5	Input Capture Flag A5 0: No input capture has occurred 1: Input capture has occurred
4	ICFA4	Input Capture Flag A4 0: No input capture has occurred 1: Input capture has occurred
3	ICFA3	Input Capture Flag A3 0: No input capture has occurred 1: Input capture has occurred
2	ICFA2	Input Capture Flag A2 0: No input capture has occurred 1: Input capture has occurred
1	ICFA1	Input Capture Flag A1 0: No input capture has occurred 1: Input capture has occurred
0	ICFA0	Input Capture Flag A0 0: No input capture has occurred 1: Input capture has occurred

The timer status register A (TSRA) is an 8-bit read-only register that indicates occurrence of overflow of the free-running counter A (TCNTA) and input capture of input capture registers A0 to A6 (ICRA0 to ICRA6).

These status flags indicate occurrence of an interrupt request, and can be cleared to 0 by setting the corresponding bit in timer status clear register A (TSCRA). When an interrupt source occurs while one of these flags is set to 1, an interrupt request is generated again. Even if flag clear by the corresponding timer status clear register conflicts with flag setting due to occurrence of an interrupt source, an interrupt request is generated.

TSRA is initialized to 00_H by a reset.

(1) OVFA — Overflow Flag A

Indicates that free-running counter A (TCNTA) has overflowed.

This flag indicates that an overflow occurred in TCNTA when this bit are read as 1. This flag can be set neither to 1 nor 0 by software.

- Setting (to 1) condition
  When TCNTA overflows (transition from FFFF $FFFF_H$ to 0000 0000_H)
- Clearing (to 0) condition
  When OVFCA in timer status clear register A (TSCRA) is set to 1

(2) ICFA0 to 6 — Input Capture Flag A0 to 6

These bits indicate that the value in free-running counter A (TCNTA) has been captured by input capture register A0 to 6 (ICRA0 to ICRA6). When one of these bits is read as 1, the value in TCNTA has been stored in the corresponding ICRA.

These bits can be set neither to 1 nor 0 by software.

- Setting (to 1) condition
  When TCNTA is transferred to ICRA0 to ICRA6 on assertion of the input capture signal (TIA00 to TIA06)
- Clearing (to 0) condition
  When ICFCA0 to ICFCA6 in timer status clear register A (TSCRA) are set to 1

21.4.2.10 TSCRA — Timer Status Clear Register A

Access: 8-bit accessible
Writing 0 is disabled. If 1 is written to this register, data is not retained. This register is always read as 0.

Address: FFE6 020A$_H$

Value after reset: 00$_H$

Bit	7	6	5	4	3	2	1	0
	OVFCA	ICFCA6	ICFCA5	ICFCA4	ICFCA3	ICFCA2	ICFCA1	ICFCA0
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.20 TSCRA Register Contents

Bit Position	Bit Name	Function
7	OVFCA	Overflow Flag Clear A Enable 0: Disabled (Initial value) 1: OVFA in the timer status register A (TSRA) is cleared to 0.
6	ICFCA6	Input Capture Flag Clear A6 Enable 0: Disabled (Initial value) 1: ICFA6 in the timer status register A (TSRA) is cleared to 0.
5	ICFCA5	Input Capture Flag Clear A5 Enable 0: Disabled (Initial value) 1: ICFA5 in the timer status register A (TSRA) is cleared to 0.
4	ICFCA4	Input Capture Flag Clear A4 Enable 0: Disabled (Initial value) 1: ICFA4 in the timer status register A (TSRA) is cleared to 0.
3	ICFCA3	Input Capture Flag Clear A3 Enable 0: Disabled (Initial value) 1: ICFA3 in the timer status register A (TSRA) is cleared to 0.
2	ICFCA2	Input Capture Flag Clear A2 Enable 0: Disabled (Initial value) 1: ICFA2 in the timer status register A (TSRA) is cleared to 0.
1	ICFCA1	Input Capture Flag Clear A1 Enable 0: Disabled (Initial value) 1: ICFA1 in the timer status register A (TSRA) is cleared to 0.
0	ICFCA0	Input Capture Flag Clear A0 Enable 0: Disabled (Initial value) 1: ICFA0 in the timer status register A (TSRA) is cleared to 0.

TSCRA is an 8-bit readable/writable register to set clearing of flags at occurrence of an overflow or input capture.

TSCRA can be read and written only in 8-bit units, but this register is always read as 0.

TSCRA is initialized to 00$_H$ by a reset.

(1) OVFCA — Overflow Flag Clear A Enable

Writing 1 to this bit while overflow flag A (OVFA) in the timer status register A (TSRA) is set to 1 clears OVFA to 0. This bit is always read as 0.

(2) ICFCA6 — Input Capture Flag Clear A6 Enable

Writing 1 to this bit while input capture flag A6 (ICFA6) in the timer status register A (TSRA) is set to 1 clears ICFA6 to 0. This bit is always read as 0.

(3) ICFCA5 — Input Capture Flag Clear A5 Enable

Writing 1 to this bit while input capture flag A5 (ICFA5) in the timer status register A (TSRA) is set to 1 clears ICFA5 to 0. This bit is always read as 0.

(4) ICFCA4 — Input Capture Flag Clear A4 Enable

Writing 1 to this bit while input capture flag A4 (ICFA4) in the timer status register A (TSRA) is set to 1 clears ICFA4 to 0. This bit is always read as 0.

(5) ICFCA3 — Input Capture Flag Clear A3 Enable

Writing 1 to this bit while input capture flag A3 (ICFA3) in the timer status register A (TSRA) is set to 1 clears ICFA3 to 0. This bit is always read as 0.

(6) ICFCA2 — Input Capture Flag Clear A2 Enable

Writing 1 to this bit while input capture flag A2 (ICFA2) in the timer status register A (TSRA) is set to 1 clears ICFA2 to 0. This bit is always read as 0.

(7) ICFCA1 — Input Capture Flag Clear A1 Enable

Writing 1 to this bit while input capture flag A1 (ICFA1) in the timer status register A (TSRA) is set to 1 clears ICFA1 to 0. This bit is always read as 0.

(8) ICFCA0 — Input Capture Flag Clear A0 Enable

Writing 1 to this bit while input capture flag A0 (ICFA0) in the timer status register A (TSRA) is set to 1 clears ICFA0 to 0. This bit is always read as 0.

21.4.2.11 ICRAx — Input Capture Registers Ax (x = 0 to 6)

Access: 32-bit accessible but 8-bit/16-bit inaccessible

Address: FFE6 0240_H (ICRA0)
FFE6 0260_H (ICRA1)
FFE6 0280_H (ICRA2)
FFE6 $02A0_H$ (ICRA3)
FFE6 $02C0_H$ (ICRA4)
FFE6 $02E0_H$ (ICRA5)
FFE6 $0300H_H$ (ICRA6)

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ICAx															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ICAx															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 21.21 ICRAx Register Contents

Bit Position	Bit Name	Function
31 to 0	ICAx	Input Capture Ax These bits hold 32-bit input capture value.

Note: x is an integer of 0 to 6.

Input capture registers A0 to A6 (ICRA0 to ICRA6) are 32-bit read-only registers only for input capture. No value can be written to these registers.

Input capture-dedicated registers detect an external input capture signal (TIA00 to TIA06) and store the free-running counter A (TCNTA) value. The DMAC can be activated at this input capture timing. At this time, the corresponding bit in the timer status register A (TSRA) is set to 1.

The input capture signal edge to be extracted is set by the I/O control bits in the timer I/O control register 1A (TIOR1A).

ICRA0 to ICRA6 can be read only in 32-bit units. Do not read these registers in 16-bit or 8-bit units.

ICRA0 to ICRA6 are initialized to 0000 0000_H by a reset.

21.4.2.12 TCNTA — Free-Running Counter A

Access: 32-bit accessible but 8-bit/16-bit inaccessible

Address: FFE6 0220$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CNTA															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CNTA															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.22 TCNTA Register Contents

Bit Position	Bit Name	Function
31 to 0	CNTA	Timer Count A These bits hold the counter value in 32-bit units.

TCNTA is a 32-bit readable/writable register that counts on the signal output by the prescaler via the clock bus, externally input clock signal, or multiplied-and-corrected signal output by timer B.

Timer A is started for counting up by setting the TAE bit in the ATU-IV master enable register (ATUENR) to 1. The clock input to the counter is selected by setting the clock select bit (CKSELA) in timer control register 1A (TCR1A).

When TCNTA overflows (FFFF FFFF$_H$ to 0000 0000$_H$), an overflow interrupt is generated and the overflow flag (OVFA) in timer status register A (TSRA) is set to 1.

TCNTA can be read and written only in 32-bit units. Do not read or write TCNTA in 16-bit or 8-bit units.

TCNTA is initialized to 0000 0000$_H$ by a reset.

21.4.2.13 TILRA — Timer Input Signal Level Register A

Access: 8-bit accessible

Address: FFE6 0218_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	TIAL06	TIAL05	TIAL04	TIAL03	TIAL02	TIAL01	TIAL00
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 21.23 TILRA Register Contents

Bit Position	Bit Name	Function
7	—	This bit is not used. Fix this bit to 0.
6 to 0	TIALx	Pin Level Read Register only for reading the TIA0x pin level

The timer input signal level register A (TILRA) is an 8-bit read-only register that can read external input (TIA00 to TIA06) pin levels after noise cancellation.

TILRA can be read only in 8-bit units.

TILRA is initialized to 00_H by a reset.

21.4.2.14 TILCRA — Timer Input Signal Level Capture Register A

Access: 8-bit accessible

Address: FFE6 021A$_H$

Value after reset: 00$_H$

Bit	7	6	5	4	3	2	1	0
	—	CTIAL06	CTIAL05	CTIAL04	CTIAL03	CTIAL02	CTIAL01	CTIAL00
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 21.24 TILCRA Register Contents

Bit Position	Bit Name	Function
7	—	This bit is not used. Fix this bit to 0.
6 to 0	CTIAL0x	Capture Pin Level Register only for reading the TIA0x pin level capture

The timer input signal level capture register A (TILCRA) is an 8-bit read-only register that can capture and hold external input signal pin (TIA00 to TIA06) levels after noise cancellation.

Pin level is captured when TCNTB6 of timer B matches OCRB6 regardless of the setting of the IREGB6 bits in TICRB of timer B.

TILCRA can be read only in 8-bit units.

TILCRA is initialized to 00$_H$ by a reset.

21.4.2.15 NCNTAx — Noise Canceler Counters Ax (x = 0 to 6)

Access: 16-bit accessible but 8-bit inaccessible

Address: FFE6 0244_H (NCNTA0)
FFE6 0264_H (NCNTA1)
FFE6 0284_H (NCNTA2)
FFE6 $02A4_H$ (NCNTA3)
FFE6 $02C4_H$ (NCNTA4)
FFE6 $02E4_H$ (NCNTA5)
FFE6 0304_H (NCNTA6)

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	NCNTAx															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.25 NCNTA Register Contents

Bit Position	Bit Name	Function
15 to 0	NCNTAx	Noise Canceler Count Ax 16-bit count value

Note: x is an integer of 0 to 6.

NCNTA0 to NCNTA6 are 8-bit readable/writable registers that are started for counting up by an assertion of externally input signals (TIA00 to TIA06) as a trigger when the noise cancelers are enabled by setting the noise canceler enable bit (NCEA6 to NCEA0) in timer I/O control register 2A (TIOR2A) to 1. In level accumulation cancellation mode, these registers increment or decrement according to the external input level. The count clock for the noise cancelers or clock bus 5 is selectable as a count source by the noise canceler clock select bits (NCKA0 to NCKA6) in TIOR2A.

NCNTA0 to NCNTA6 can be read and written only in 16-bit units.

NCNTA0 to NCNTA6 are initialized to 00_H by a reset.

According to the settings of the noise cancellation mode select bit (NCMSEL) and the timer A noise cancellation mode bit (NCMA) in the noise cancellation mode register (NCMR) in the common control unit, the noise cancellation mode bits (NCM1Ax) in noise cancellation mode channel register 1A (NCMCR1A) of timer A, and the noise cancellation mode bits (NCM2Ax) in noise cancellation mode channel register 2A (NCMCR2A), these registers operate in premature-transition cancellation mode, minimum time-at-level cancellation mode, or level accumulation cancellation mode.

- Premature-transition cancellation mode

  When a level change of the externally input signal (TIA00 to TIA06) is detected while bits NCEA0 to NCEA6 are set to 1 and NCNTA0 to NCNTA6 are stopped, NCNTA0 to NCNTA6 are started for counting up. These counters are cleared to 0000_H and stopped on the first edge of the PCLK after the value in NCNTA0 to NCNTA6 matches the value in noise canceler registers A0 to A6 (NCRA0 to NCRA6).

  NCNTA0 to NCNTA6 are incremented regardless of the TAE bits in the ATU-IV master enable register (ATUENR).

The first change is output as the signal whose noise is removed and the edge is to be extracted. Subsequent level changes are masked until the value in the counter reaches the value in the noise canceler register (NCRA0 to NCRA6). The level of the externally input signal is output as the signal following noise cancellation at the time of the match in comparison.

When the NCEA bits are cleared to 0 while the counter (NCNTA0 to NCNTA6) is being incremented, counting continues until the values in the counter and the noise canceler register match. The subsequent level changes are also masked over this period.

- Minimum time-at-level cancellation mode

  When a level change of the externally input signal (TIA00 to TIA06) is detected while bits NCEA6 to NCEA0 are set to 1 and NCNTA0 to NCNTA6 are stopped, NCNTA0 to NCNTA6 are started for counting up. These counters are cleared to 0000_H and stopped on the first edge of the PCLK after the value in NCNTA0 to NCNTA6 matches the value in noise canceler registers A0 to A6 (NCRA0 to NCRA6) or after the level of the externally input signal (TIA00 to TIA06) is changed during count operation.

  NCNTA0 to NCNTA6 are incremented regardless of the TAE bits in the ATU-IV master enable register (ATUENR).

  When a change in the level of an externally input signal is detected, the corresponding noise canceler counter (NCNTA0 to NCNTA6) starts counting up. If subsequent level changes are not detected until the value in the counter reaches the value in the noise canceler register (NCRA0 to NCRA6), the previously accepted level change is output as the signal following noise cancellation at the time of the match in comparison between the counter and noise canceler register. When a subsequent level change is detected before the values in the counter and noise canceler register match, the signal is treated as noise. The output signal from which noise has been removed is not changed in this case.

  When the NCEA bits are cleared to 0 while the counter (NCNTA0 to NCNTA6) is being incremented, counting continues until the values in the counter and the noise canceler register match or a change in the level of the externally input signal is detected.

- Level accumulation cancellation mode

  While NCEA0 to NCEA6 bits are set to 1, NCNTA0 to NCNTA6 increment or decrement according to the input signal level. When the input level is high, NCNTA0 to NCNTA6 increment. When the counter value matches the noise canceler register A0 to A6 (NCRA0 to NCRA6) value, up-counting stops on the next PCLK. When the input level is low, NCNTA0 to NCNTA6 decrement. When the counter value reaches 0000_H, down-counting stops on the next PCLK.

  NCNTA0 to NCNTA6 count regardless of the TAE bit value in the ATU-IV master enable register (ATUENR).

  When the counter value matches the NCRA0 to NCRA6 value during up-counting, the noise canceler output is updated to 1. When the counter value reaches 0000_H during down-counting, the noise canceler output is updated to 0.

  When the NCEA bit is cleared during counting, the noise canceler counter stops working and the value changes from the noise canceler output to the input signal level at that time.

  For this reason, note that when clearing the NCEA bit in level accumulation cancellation mode, edge detection may be made at this changeover.

21.4.2.16 NCRAx — Noise Canceler Registers Ax (x = 0 to 6)

Access: 16-bit accessible but 8-bit inaccessible

Address: FFE6 0246$_H$ (NCRA0)
FFE6 0266$_H$ (NCRA1)
FFE6 0286$_H$ (NCRA2)
FFE6 02A6$_H$ (NCRA3)
FFE6 02C6$_H$ (NCRA4)
FFE6 02E6$_H$ (NCRA5)
FFE6 0306$_H$ (NCRA6)

Value after reset: 0000$_H$

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	NCTAx															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.26 NCRAx Register Contents

Bit Position	Bit Name	Function
15 to 0	NCTAx	Noise Canceler Count Ax TIA0x noise cancellation period (16-bit comparison value)

Note: x is an integer of 0 to 6.

NCRA0 to NCRA6 are 16-bit readable/writable registers that set the upper limitations of noise cancel counters Ax (NCNTA0 to NCNTA6). For example, when the noise canceler is driven by the PCLK divided by 128 and these registers are set to FFFF$_H$, a pulse whose width is 0.21 s (when PCLK is 40 MHz) can be treated as noise.

According to the settings of the noise cancellation mode select bit (NCMSEL) and the timer A noise cancellation mode bit (NCMA) in the noise cancellation mode register (NCMR) in the common control unit, the noise cancellation mode bits (NCM1Ax) in noise cancellation mode channel register 1A (NCMCR1A) of timer A, and the noise cancellation mode bits (NCM2Ax) in noise cancellation mode channel register 2A (NCMCR2A), these registers operate in premature-transition cancellation mode, minimum time-at-level cancellation mode, or level accumulation cancellation mode.

- Premature-transition cancellation mode

  While NCNTAx is in count operation, the level change of the subsequent input signal is masked. Values of NCNTAx and NCRAx are constantly compared. When these values match, the NCNTAx count value is cleared on the next PCLK, the counter stops, and the input signal mask is canceled.

- Minimum time-at-level cancellation mode

  While NCNTA0 to NCNTA6 are in count operation, noise canceler processing waiting state is entered. Values in NCNTA0 to NCNTA6 and NCRA0 to NCRA6 are constantly compared. When these values match, the NCNTA0 to NCNTA6 count value is cleared on the next PCLK, the counter stops, and at the same time the noise canceler outputs the input signal that has passed through the noise canceling processing.

- Level accumulation cancellation mode

  While NCNTA0 to NCNTA6 are in up-count operation, values of NCNTA0 to NCNTA6 and NCRA0 to NCRA6 are compared. When these values match, the NCNTA0 to NCNTA6 stop up-counting on the next PCLK. While NCNTA0 to NCNTA6 are in down-count operation, the NCNTA0 to NCNTA6 value is compared with 0000_H.

  NCRA0 to NCRA6 can be read and written only in 16-bit units.

  NCRA0 to NCRA6 are initialized to 0000_H by a reset.

21.4.3 Detailed Operation

21.4.3.1 Operation of Noise Canceler

The noise canceller has three operating modes. Settings for the noise cancellation mode register (NCMR) in the common controller and of noise cancellation mode channel register 1A (NCMCR1A) and noise cancellation mode channel register 2A (NCMCR2A) of timer A select premature-transition cancellation mode, minimum time-at-level cancellation mode, or level accumulation cancellation mode.

Figure 21.7 shows an example of operations for noise cancellation in premature-transition cancellation mode, and **Figure 21.9** shows an example of operations for noise cancellation in minimum time-at-level cancellation mode. These figures show the situation where the settings are for detection of edges and for the edges to be rising in the signal input to TIA00.

In premature-transition cancellation mode, counting by noise cancellation counter Ax (NCNTAx) starts in response to a change in the level of the input signal as a trigger. At the same time as counting starts, its change in the level of the input signal is output as the signal after noise cancellation.

Counting continues until the value in the counter matches that in noise cancellation register Ax (NCRAx). All changes in the level of the input signal are ignored over this period and are not output in the signal following processing for noise cancellation.

The level being input at the time the value of the counter has matched that of the noise cancellation register is then output as the signal following processing for noise cancellation. Accordingly, note that the levels following processing for noise cancellation are changed on the compare match when the input levels at the start of counting (level following a change of level) and on the compare match differ.

Figure 21.8 shows an example of noise cancellation for two type of input waveforms in premature-transition cancellation mode.

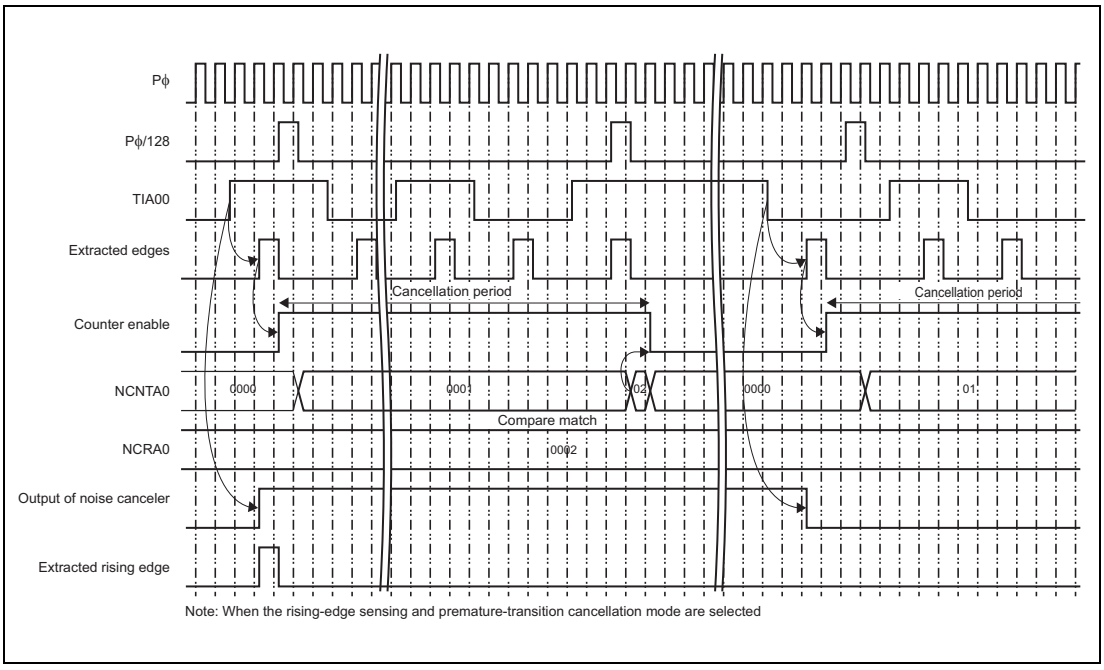


Figure 21.7 Example of Noise Cancellation in Premature-Transition Cancellation Mode

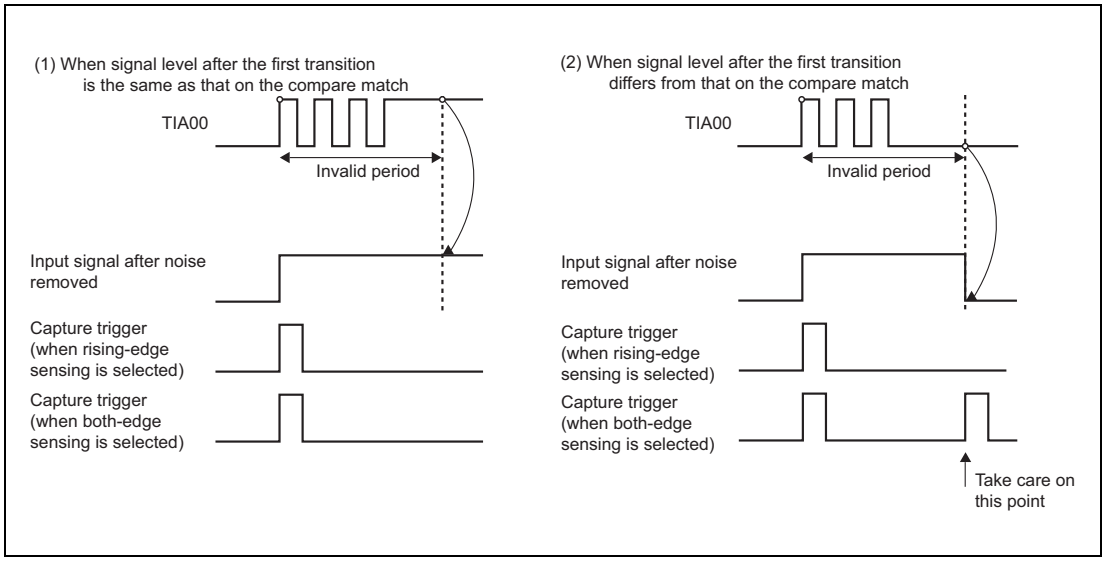


Figure 21.8 Example of Noise Cancellation in Premature-Transition Cancellation Mode for Two Types of Input Waveforms

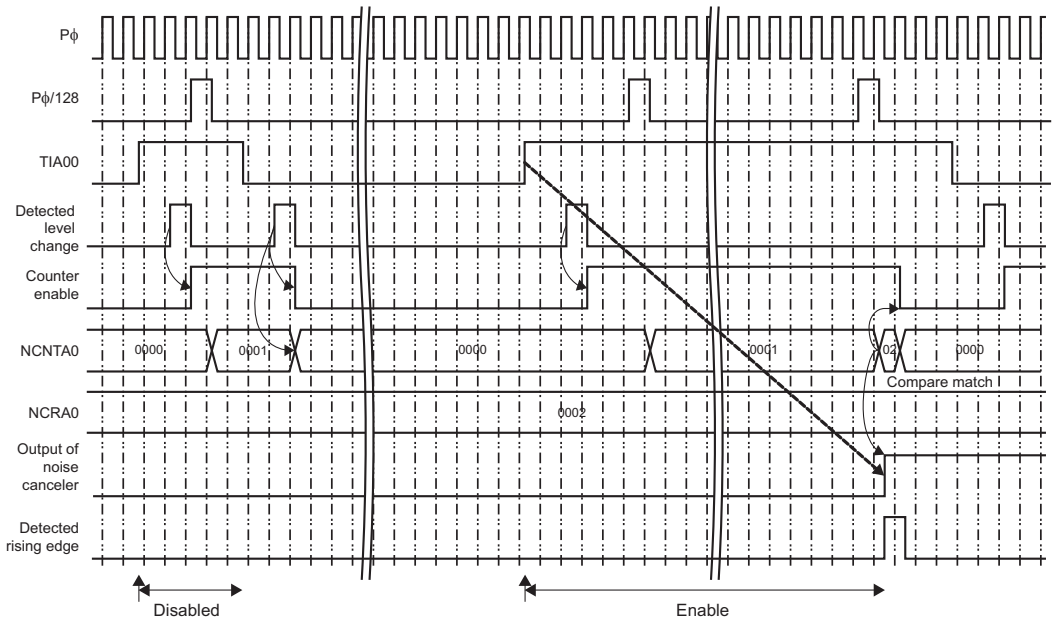
In minimum time-at-level cancellation mode, counting by noise cancellation counter Ax (NCNTAx) starts in response to a change in the level of the input signal as a trigger. Counting continues until the value counted matches that of the noise cancellation register Ax (NCRAx) or until the level of the input signal changes.

When the value of the counter matches that of the noise cancellation register, a change in the level of the input at the start of counting is output as the signal following noise cancellation. If the level of the input signal changes before the values of the counter and of the noise cancellation register match, its change and the change of level from that when counting started are deemed to be noise, and are not output in the signal following noise cancellation.

Figure 21.10 shows an example of shows an example of operations for noise cancellation in level accumulation cancellation mode. These figures show the situation where the settings are for detection of edges and for the edges to be rising in the signal input to TIA00.

In level accumulation cancellation mode, whether noise cancellation counter Ax (NCNTAx) counts up or down depends on the level of the input signal. If the counter is counting up, this continues until the value counted matches that of noise cancellation register Ax (NCNTAx), or if it is counting down, this continues until the value counted matches 0000_H.

Updating of the output of the noise canceller proceeds when there are matches in comparison. When the value of the counter matches that of NCRAx during up-counting, the noise canceller output value is updated to 1, and when the value of the counter matches 0000_H during down-counting, the noise canceller output value is updated to 0.



Note: Rising-edge sensing and minimum time-at-level cancellation mode are selected in this example.

Figure 21.9 Example of Noise Cancellation in Minimum Time-at-Level Cancellation mode

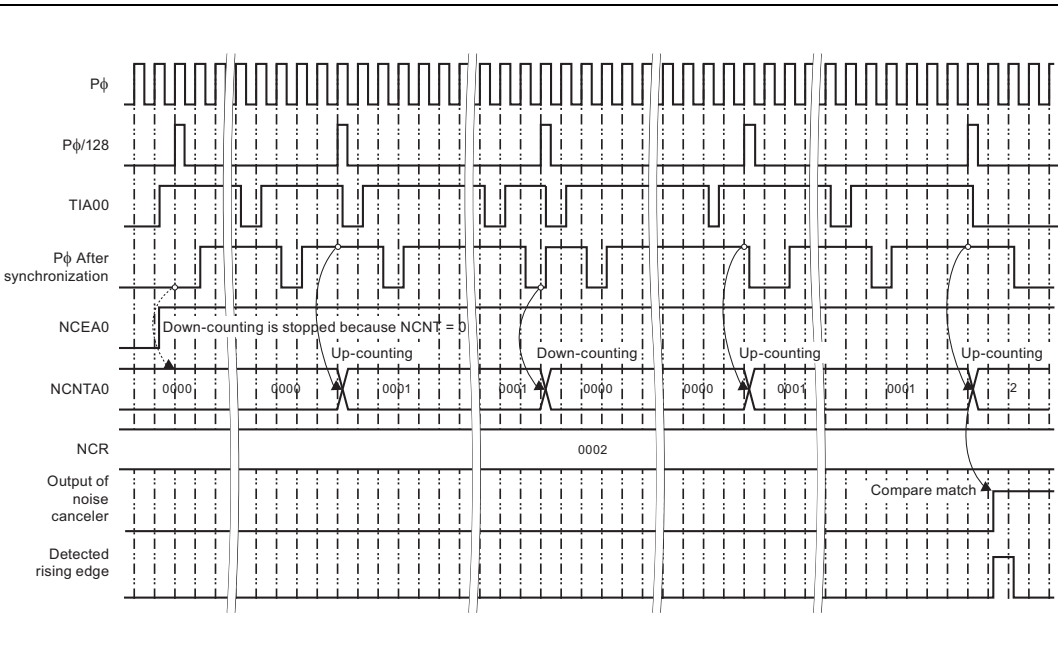


Figure 21.10 Example of Noise Cancellation in Level Accumulation Cancellation Mode

21.4.3.2 Operation of Free-Running Counter

Free-running counter A (TCNTA) is started for counting up by setting the TAE bit in ATU-IV master enable register (ATUENR) to 1. When TCNTA overflows (from FFFF FFFF$_H$ to 0000 0000$_H$), the OVFA bit in timer status register A (TSRA) is set to 1. An interrupt request is issued to the CPU at the time. After overflow, TCNTA continues counting up from 0000 0000$_H$.

When the TAE bit in ATU-IV master enable register (AUTENR) is cleared to 0, TCNTA is stopped but is not cleared. By setting the TAE bit to 1 again, TCNTA is resumed from the value when stopped.

TCNTA can be written during operation and writing takes priority over counting. After that, TCNTA is started from the written value. Regardless of the clock driving the counter, the write access is completed in two cycles of the PCLK clock.

CAUTION

The prescalers run regardless of the TAE bit and are not synchronized with the timing at which the TAE bit is set. Therefore, the time from when the TAE bit is set to when TCNTA is incremented for the first time is less than the cycle of the clock of TCNTA.

Figure 21.11 shows an example of free-running counter A (TCNTA) operation.

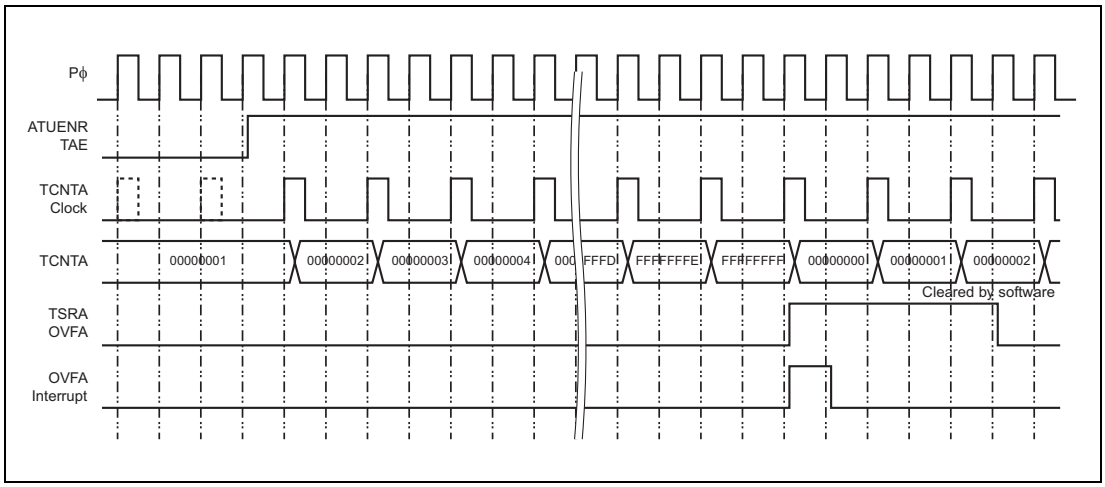


Figure 21.11 Example of Free-Running Counter A (TCNTA) Operation: Overflow Timing

21.4.3.3 Input Capture

Input capture is performed by input capture registers A0 to A6 (ICRA0 to ICRA6) when input capture is enabled in bits IOA6 to IOA0 in timer I/O control register 1A (TIOR1A). ICRA0 to ICRA6 capture the value in free-running counter A (TCNTA) by an edge of externally input signals (TIA00 to TIA06).

Noise on the signals can be removed by the noise cancelers. TCNTA is started for counting up by setting the TAE bit in ATU-IV master enable register (ATUENR). When an edge is input on the corresponding signal, the bit in timer status register (TSRA) is set to 1 and the value in TCNTA is transferred to ICRA. The rising or falling edge, or both edges can be selected. At this time, it is also possible to output an interrupt request to the CPU, and start DMA transfer.

When input capture registers and free-running counter A (TCNTA) are written simultaneously, ICRA0 to ICRA6 capture the previous value stored in TCNTA.

Figure 21.12 shows an example of input capture when the edges to be sensed are rising edges for TIA00, falling edges for TIA01, and both edges for TIA02.

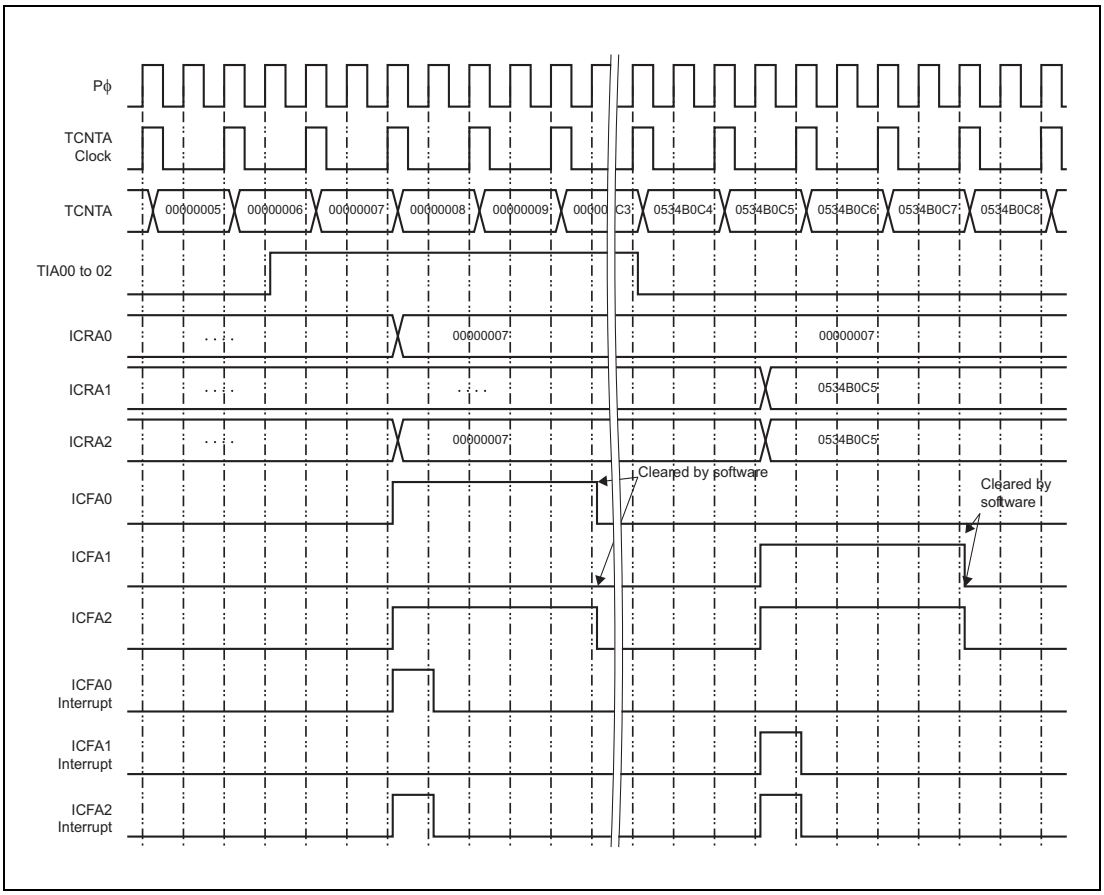


Figure 21.12 Example of Input Capture of Timer A

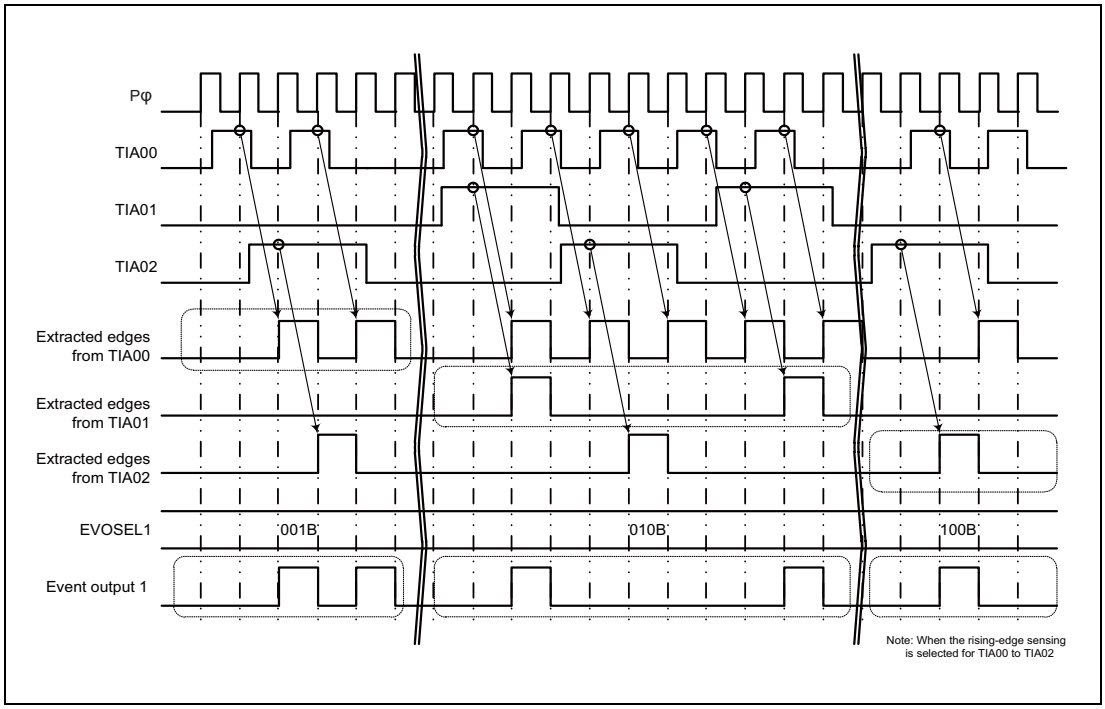


Figure 21.13 Example of TIA00 to TIA02 Event Output

By setting the EVOSEL1 bit in timer control register 1A (TCR1A), any one of edges of external input pins TIA00 to TIA02 can be detected and signals after noise cancellation can be output to timer B and timer C as an event. Furthermore, either TIA01 or TIA02 (signal after edge detection and noise cancellation) can be output to timer C and timer D as an event by setting the EVOSEL2 bit and EVOSEL2B bit. In addition, any one of external input pins TIA00 to TIA06 or a combination of them (signals after edge detection and noise cancellation) can be output as an event to timer B and timer C by setting the EVOSELE1E and EVOSELE1 bits in timer control register 2A (TCR2A). Any one of external input pins TIA00 to TIA06 or a combination of them (signals after edge detection and noise cancellation) can be output as an event to timer C and timer D by setting the EVOSELE2AE and EVOSELE2A bits in timer control register 3A (TCR3A) and the EVOSELE2BE and EVOSELE2B bits in timer control register 4A (TCR4A).

Figure 21.13 shows an example of event output operation in case EVOSEL1 is set to 001_B (TIA00 input edge output), 010_B (TIA01 input edge output), and 100_B (TIA02 input edge output).

21.4.3.4 Pin Level Capture Operation

Levels of the external input pins (TIA00 to TIA06) after noise cancellation can be read from the timer input signal level register A (TILRA). Furthermore, the value of the timer input signal level registerA (TILRA) can be captured to the timer input signal level capture register A (TILCRA). The value is captured from TILRA to TILCRA by the missing-tooth detection trigger at the timing of compare match between TCNTB6 and OCRB6 of timer B. **Figure 21.14** shows an example of operation of the TILRA register and the TILCRA capture triggered by missing-tooth detection when signals are input to TIA00 to TIA06.

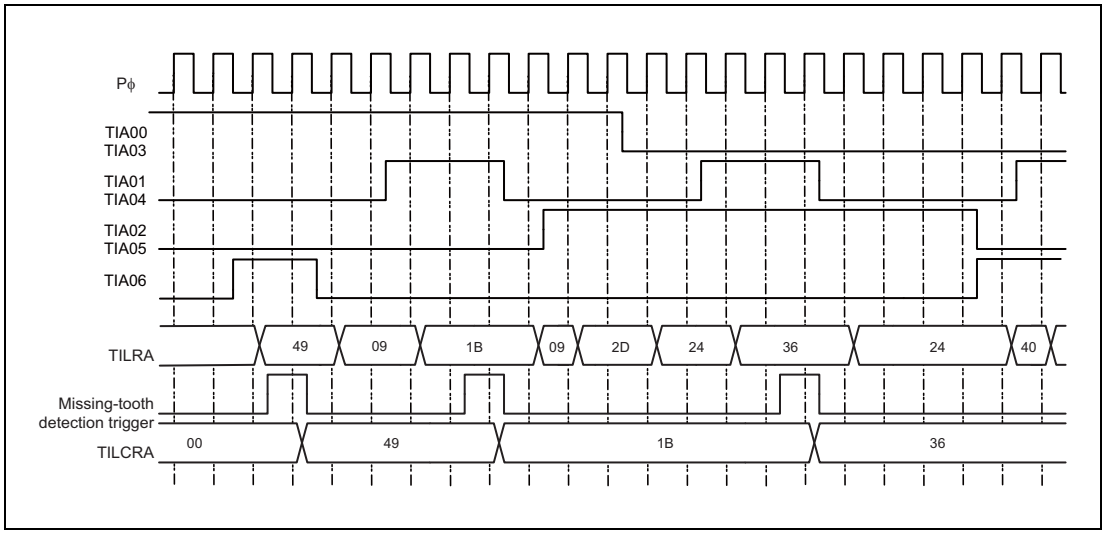


Figure 21.14 Example of TILCRA Capture Operation

21.4.3.5 DMA Transfer

Setting the DMAC allows it to be activated by input capture operation of timer A.

21.5 Timer B

21.5.1 Operation

Timer B generates a multiplied-and-corrected clock signal based on an external-event input and supplies the generated signal to other timers. Timer B consists of an edge-interval measuring block, frequency-multiplying clock generator, and corrector for the frequency-multiplied clock signal.

Edge-Interval Measuring Block

The edge-interval measuring block measures the intervals between edges of external-event signals input via timer A (TCNTB0, ICRB0). Interrupt requests can be issued for the CPU in response to matches between edge interval measuring counter B0 (TCNTB0) and output compare register B0 (OCRB0). TCNTB0 can also be captured in ICRB2 via ICRB1 on matches between event counter B1 (TCNTB1) and output compare register B1 (OCRB1). Counting by TCNTB1 is driven by the external-event input. This provides a way to measure the intervals between multiple event inputs. Although TCNTB0 is cleared every event input, ICRB1 keeps a running total of the TCNTB0 value. ICRB2 latches the running totals on compare matches of the event counter. Furthermore, it is also possible to set the value of the event counter B1 (TCNTB1) that uses an external event input as a count source to 01_H by the CLRB1 bit(TCNTB1 clear bit) in the timer control register B (TCRB) at the next external event input timing after an event compare match with OCRB10. Furthermore, up to seven capture values (ICRB0) can be retained in record registers B-1 to 6 (RECRB1 to RECRB6) at the external input event timing. Values retained in ICRB0 and RECRB1 to RECRB6 can be backed up in record backup registers B 0 to 6 (RBURB0 to RBURB6) on event compare match between the event counter B1 (TCNTB1) and OCRB12.

In the edge-interval measuring block, the event counter B1(TCNTB1) value is captured for the event counter B1(TCNTB1) at occurrence of any of seven external event input signals 1_B to 1_H that are input via timer A (ICRB30 to ICRB36). The event counter B1(TCNTB1) is not cleared at occurrence of any of external event input signals 1_B to 1_H.

Frequency-Multiplied Clock Generator

The frequency-multiplied clock generator generates a clock signal obtained by frequency-multiplying an external-event input signal by a value from 1 to 4095.

In the edge-interval measuring block, down-counting (TCNTB2) with reload (RLDB) is performed for the capture value (ICRB0) in the input edge-interval measuring counter B0 (TCNTB0). Down-counting is performed by the set division ratio value specified in the PIMR select register. In addition to PIMR1, a division ratio value of PIMR2 can be selected by an event compare match between the event counter B1 (TCNTB1) in the edge-interval measuring block and OCRB11. When the down counter underflows, the multiplied clock (AGCK1) signal is asserted.

TCNTB6 is a measuring counter using the multiplied clock as a source clock, and an interrupt request to the CPU can be output due to the compare match with OCRB6. ICRB6 is a capture register, and this TCNTB6 value is captured when an external event input occurs.

TCNTB6M is a measuring counter using the multiplied clock as a source clock. The up-count value can be set within a range from 0 to less than 4 in units of 1/64. When an external event input occurs, the TCNTB6M value is compared with the ICRB6 value. When the TCNTB6M value is smaller than the ICRB6 value, an interrupt request to the CPU can be output.

Frequency-Multiplied Clock Signal Corrector

The multiplied clock signal (AGCK1) needs to be corrected when two consecutive edge intervals differ significantly, since the earlier interval is referred to in calculating the multiplier for the current interval. The frequency-multiplied clock signal corrector generates a multiplied-and-corrected clock signal (AGCKM) by using three correcting counters (TCNTB3 to TCNTB5) and correcting counter clearing register B (TCCLRB). Output of the clock signal thus produced on clock-bus line 5 can be selected by using the clock-bus control register (CBCNT). Other timers can then use this clock as a source for counting. For the automatic up-counting function that continues counting until the multiplied-and-corrected clock generating counter B5 (TCNTB5) value matches the correcting counter clear register B (TCCLRB) value when the multiplied-and-corrected clock (AGCKM) is generated, whether to enable or disable this function is selectable. Furthermore, the correcting counter B3 (TCNTB3) can be automatically cleared by the CLRB3 bit (TCNTB3 clear bit) in the timer control register B (TCRB) at the timing of an event compare match between the event counter B1 (TCNTB1) in the edge-interval measuring block and OCRB10 or between the correcting event counter B3 (TCNTB3) and OCRB8.

Setting ACRTRG in the AGCKM2 correction enable setting register (ACRTRGB) to 1 corrects the multiplied-and-corrected clock (AGCKM). The multiplied-and-corrected clock 2 (AGCKM2) corrected from AGCKM is output by operation of the internal AGCKM2 counter according to the AGCKM2 correction clock count setting register (ACRVALRB).

The internal AGCKM2 counter keeps operating even in the break state caused by the debugger during counting. To stop counting of the internal AGCKM2 counter, set ACRCLR in the AGCKM2 correction clear setting register (ACRCLRB) to 1.

Figure 21.15 is a block diagram of timer B.

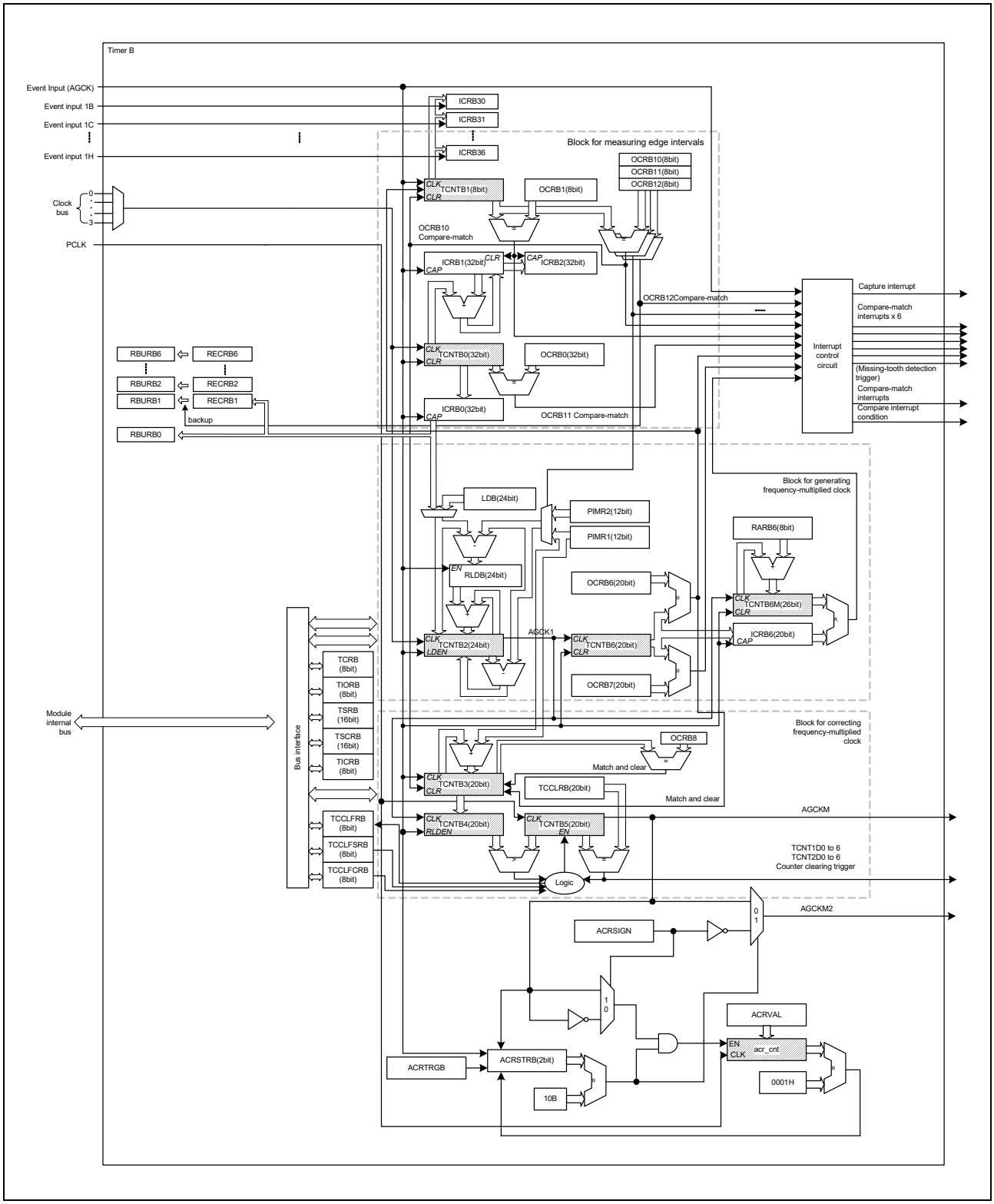


Figure 21.15 Timer B Block Diagram

21.5.2 Timer B Control Registers

21.5.2.1 TCRB — Timer Control Register B

Access: 8-bit accessible

Address: FFE6 0400_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	PIMRSEL	CLRB3	CLRB1	CLRB3SEL		CLRB1SEL	CKSELB	
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.27 TCRB Register Contents

Bit Position	Bit Name	Function
7	PIMRSEL	PIMR Select 0: PIMR1 is always selected. 1: PIMR2 is selected instead of PIMR1 while TCNTB1 and OCRB11 match.
6	CLRB3	TCNTB3 Clear 0: TCNTB3 is not cleared. 1: TCNTB3 is cleared.
5	CLRB1	TCNTB1 Clear 0: TCNTB1 is not cleared 1: TCNTB1 is cleared.
4, 3	CLRB3SEL	TCNTB3 Clear Select 00: Compare match between TCNTB1 and OCRB10 is used to clear TCNTB3. 01: Compare match between TCNTB6 and OCRB6 is used to clear TCNTB3. 10: Compare match between TCNTB3 and OCRB8 is used to clear TCNTB3. 11: Setting prohibited
2	CLRB1SEL	TCNTB1 Clear Select 0: Compare match between TCNTB1 and OCRB10 is used to clear TCNTB1. 1: Compare match between TCNTB6 and OCRB6 is used to clear TCNTB1.
1, 0	CKSELB	Block Select B 00: Clock-bus line 0 is selected 01: Clock-bus line 1 is selected 10: Clock-bus line 2 is selected 11: Clock-bus line 3 is selected

TCRB is an 8-bit readable/writable register that allows you to select a count source for the input edge interval measuring counter B0 (TCNTB0) and the reload counter B2 (TCNTB2), select whether to clear the TCNTB1 and TCNTB3 values at the time of a match in comparison between TCNTB1 and OCRB10 or of that between TCNTB6 and OCRB6, and to select a pulse interval multiplier.

TCRB can be read and written in 8-bit units.

TCRB is initialized to 00_H by a reset.

(1) CKSELB — Clock Select B

These bits select the clock driving TCNTB0 and TCNTB2 from clock-bus lines 0 to 3, which are clock signals divided by prescalers 0 to 3. The counters are incremented on the rising edge of the selected clock.

To select the clock, stop timer B operation.

(2) PIMRSEL — Pulse Interval Multiplier Select

By setting the PIMRSEL bit to 1 while TCNTB1 matches OCRB11, PIMR2 is selected for down-count values to be loaded to reload register B (RLDB) or for down-count values of TCNT2B. In other cases, PIMR1 is selected. However, PIMR1 is always used for adding the TCNTB3 value.

While the PIMRSEL bit is 0, PIMR1 is always used.

(3) CLRB1 — TCNTB1 Clear

Setting the TCNTB1 clear bit (CLRB1) to 1 clears the TCNTB1 value at the timing of the first event input after a match in comparison between TCNTB1 and OCRB10 or between TCNTB6 and OCRB6 for the CLRB1SEL setting condition.

(4) CLRB3 — TCNTB3 Clear

Setting the TCNTB3 clear bit (CLRB3) to 1 clears the TCNTB3 value in synchronization with the first PCLK after a match in comparison between TCNTB1 and OCRB10, between TCNTB6 and OCRB6, or between TCNTB3 and OCRB8 for the CLRB3SEL setting condition.

(5) CLRB1SEL — TCNTB1 Clear Select

Setting the TCNTB1 clear select bit (CLRB1SEL) sets the TCNTB1 clearing condition. When CLRB1SEL is 0, TCNTB1 can be cleared in response to a match in comparison between TCNTB1 and OCRB10. When CLRB1SEL is 1, TCNTB1 can be cleared in response to a match in comparison between TCNTB6 and OCRB6.

(6) CLRB3SEL — TCNTB3 Clear Select

Setting the TCNTB1 clear select bit (CLRB3SEL) sets TCNTB3 clearing condition. When CLRB3SEL is 00, TCNTB3 can be cleared in response to a match in comparison between TCNTB1 and OCRB10. When CLRB3SEL is 01, TCNTB3 can be cleared in response to a match in comparison between TCNTB6 and OCRB6. When CLRB3SEL is 10, TCNTB3 can be cleared in response to a match in comparison between TCNTB3 and OCRB8.

21.5.2.2 TIORB — Timer I/O Control Register B

Access: 8-bit accessible

Address: FFE6 0402_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	LDSEL	CTCNTB5	EVCNTB	LDEN	CCS	—	—	IOB6
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W

Table 21.28 TIORB Register Contents

Bit Position	Bit Name	Function
7	LDSEL	Loading Data Select 0: ICRB0 value is used to calculate the data to be loaded to TCNTB2 and RLDB. 1: LDB value is used to calculate the data to be loaded to TCNTB2 and RLDB.
6	CTCNTB5	Count Control B5 0: TCNTB5 is in operation 1: TCNTB5 is stopped
5	EVCNTB	Event Control B 0: Disables the externally input events 1: Enables the externally input events
4	LDEN	Load Enable 0: TCNTB2 and RLDB are updated on ICEB0 input capture 1: TCNTB2 and RLDB are not updated on ICEB0 input capture
3	CCS	Counter Correction Select 0: TCNTB4 is in operation when TCNTB3 = TCNTB4 1: TCNTB4 is stopped when TCNTB3 = TCNTB4
2, 1	—	These bits are not used. Fix these bits to 0.
0	IOB6	I/O Control B6 0: Compare match between TCNTB6 and OCRB6 is disabled 1: Compare match between TCNTB6 and OCRB6 is enabled

TIORB is an 8-bit readable/writable register that selects the source of the frequency-multiplied clock and enables and disables the externally input signals, loading data, and correcting frequency multiplied clock. TIORB also controls multiplied-and-corrected clock generating counter B5 (TCNTB5) and output compare register B6 (OCRB6).

TIORB can be read and written in 8-bit units.

TIORB is initialized to 00_H by a reset.

(1) LDSEL — Loading Data Select

Selects the register to be loaded to reload counter B2 (TCNTB2) and to be used for calculating data to be loaded to reload register B (RLDB) from ICRB0 or LDB.

(2) CTCNTB5 — Count Control B5

Selects whether the multiplied-and-corrected clock generating counter B5 (TCNTB5) is enabled or stopped.

Setting this bit to 1 stops TCNTB5 and the multiplied-and-corrected clock to be output to other timers. Even if the counter is stopped, it is not cleared. Clearing this bit to 0 resumes TCNTB5 and multiplied-and-corrected clock.

(3) EVCNTB — Event Control B

Disables and enables the externally input events. Clearing this bit to 0 disables the input. Setting this bit to 1 enables the input with which input capture or generating the multiplied-and-corrected clock signal.

(4) LDEN — Load Enable

Selects whether to update the input capture register B0 (ICRB0) value, the reload counter B2 (TCNTB2) value, and the reload register B (RLDB) value at the input capture of the input edge interval measuring counter B0 (TCNTB0).

(5) CCS — Counter Correction Select

Selects whether or not correcting multiplied clock counter B4 (TCNTB4) is stopped when TCNTB3 = TCNTB4.

(6) IOB6 — I/O Control B6

Enables and disables compare match between TCNTB6 and OCRB6. When this bit is cleared to 0, compare match between TCNTB6 and OCRB6 is disabled. When it is set to 1, compare match between TCNTB6 and OCRB6 is enabled. At this time, an interrupt request is output to the CPU in response to a match in comparison.

21.5.2.3 TSRB — Timer Status Register B

Access: 8-bit and 16-bit accessible

Address: FFE6 0404_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CMFB 6E	CMFB 6M	—	CMFB 12	CMFB 11	CMFB 10	CMFB6	CMFB1	ICFB0	CMFB0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 21.29 TSRB Register Contents

Bit Position	Bit Name	Function
15 to 10	—	These bits are not used. Fix these bits to 0.
9	CMFB6E	Compare Match Flag B6E 0: Function is not selected or there are no matches in comparison. 1: A match in comparison has occurred under the condition set for IREGB6E in TICRB.
8	CMFB6M	Compare Match Flag B6M 0: The TCNTB6M value is the ICRB6 value or larger. 1: When the TCNTB6M value is smaller than ICRB6, the next event input occurs.
7	—	This bit is not used. Fix these bits to 0.
6	CMFB12	Compare Match Flag B12 0: There are no matches in comparison. 1: A match in comparison has occurred.
5	CMFB11	Compare Match Flag B11 0: No input capture has occurred. 1: Input capture has occurred.
4	CMFB10	Compare Match Flag B10 0: There are no matches in comparison. 1: A match in comparison has occurred.
3	CMFB6	Compare Match Flag B6 0: There are no matches in comparison. 1: A match in comparison has occurred.
2	CMFB1	Compare Match Flag B1 0: There are no matches in comparison. 1: A match in comparison has occurred.
1	ICFB0	Input Capture Flag B0 0: No input capture has occurred. 1: Input capture has occurred.
0	CMFB0	Compare Match Flag B0 0: There are no matches in comparison. 1: A match in comparison has occurred.

The timer status register B (TSRB) is a 16-bit read-only register that indicates occurrence of input capture and compare match. These status flags indicate occurrence of an interrupt request, and can be cleared to 0 by setting the corresponding bit in timer status clear register B (TSCRB). When an interrupt source occurs while one of these flags is set to 1, an interrupt request is generated again. Even if flag clear by the corresponding timer status clear register conflicts with flag setting due to occurrence of an interrupt source, an interrupt request is generated.

Supplement: Even if a match in comparison between TNTB6 and OCRB6 occurs while the CMFB6 flag is set, the CMFB6 bit does not generate OCRB6 compare match interrupt.

TSRB can be read and written only in 8-bit or 16-bit units.

TSRB is initialized to 0000_H by a reset.

(1) CMFB6E — Compare Match Flag B6E

This status flag indicates that both (or one of) CMFB6 and CMFB6M flags have been set by the setting for IREGB6E in the timer interrupt control register B (TICRB). When 1 is read from this flag, it shows that both (or one of) CMFB6 and CMFB6M flags have been set. This flag cannot be set to 1 or 0 by the software.

- Setting (to 1) condition
  When a CMFB6E interrupt request has occurred by the setting of IREGB6E in the timer interrupt control register B (TICRB). For details, see the description of the setting for IREGB6E in TICRB.
- Clearing (to 0) condition
  Writing 1 to CMFCB6E in timer status clear register B (TSCRB)

(2) CMFB6M — Compare Match Flag B6M

This status flag indicates that the multiplied clock counter B6M (TCNTB6M) value was smaller than the input capture register B6 (ICRB6) value at the external input event timing. When 1 is read from this flag, the TCNTB6M value is not larger than the ICRB6 value and the next event input has occurred. This flag cannot be set to 1 or 0 by the software.

- Setting (to 1) condition
  When the TCNTB6M count value does not exceed the ICRB6 value and the next event input has occurred
- Clearing (to 0) condition
  Writing 1 to CMFCB6M in timer status clear register B (TSCRB)

(3) CMFB12 — Compare Match Flag B12

This status flag indicates occurrence of a match in comparison with output compare register B12 (OCRB12). When 1 is read from this flag, it shows that a match in comparison has occurred in CMFB12. This flag cannot be set to 1 or 0 by the software.

- Setting (to 1) condition
  When the TCNTB1 count value matches the OCRB12 value and the next PCLK has occurred
- Clearing (to 0) condition
  Writing 1 to CMFCB12 in timer status clear register B (TSCRB)

(4) CMFB11 — Compare Match Flag B11

This status flag indicates occurrence of a match in comparison with output compare register B11 (OCRB11). When 1 is read from this flag, it shows that a match in comparison has occurred in CMFB11. This flag cannot be set to 1 or 0 by the software.

- Setting (to 1) condition
  When the TCNTB1 count value matches the OCRB11 value and the next PCLK has occurred
- Clearing (to 0) condition
  Writing 1 to CMFCB11 in timer status clear register B (TSCRB)

(5) CMFB10 — Compare Match Flag B10

This status flag indicates occurrence of a match in comparison with output compare register B10 (OCRB10). When 1 is read from this flag, it shows that a match in comparison has occurred in CMFB10. This flag cannot be set to 1 or 0 by the software.

- Setting (to 1) condition
  When the TCNTB1 count value matches the OCRB10 value and the next PCLK has occurred
- Clearing (to 0) condition
  Writing 1 to CMFCB10 in timer status clear register B (TSCRB)

(6) CMFB6 — Compare Match Flag B6

This status flag indicates occurrence of a match in comparison with output compare register B6 (OCRB6). When 1 is read from this flag, it shows that a match in comparison has occurred in CMFB6. This flag cannot be set to 1 or 0 by the software.

This flag is automatically cleared to 0 in response to a match in comparison with OCRB7 only when IREGB6 in TICRB is set to 10.

- Setting (to 1) condition
  When the TCNTB6 count value matches the OCRB6 value and the next AGCK1 has occurred with compare match enabled by the IOB6 bit in timer I/O control register B (TIORB)
- Clearing (to 0) condition
  - Writing 1 to CMFCB6 in timer status clear register B (TSCRB)
  - Automatically cleared to 0 in response to a match in comparison with OCRB7 when IREGB6 = 10.

(7) CMFB1 — Compare Match Flag B1

This status flag indicates occurrence of a match in comparison with output compare register B1 (OCRB1). When 1 is read from this flag, it shows that a match in comparison has occurred in CMFB1. This flag cannot be set to 1 or 0 by the software.

- Setting (to 1) condition
  When the TCNTB1 count value matches the OCRB1 value and the next PCLK has occurred
- Clearing (to 0) condition
  Writing 1 to CMFCB1 in timer status clear register B (TSCRB)

(8) ICFB0 — Input Capture Flag B0

This status flag indicates occurrence of an input capture of input capture register B0 (ICRB0). When 1 is read from this flag, it shows that an input capture has occurred in ICRB0. This flag cannot be set to 1 or 0 by the software.

- Setting (to 1) condition
  When the TCNTB0 value has been transferred to ICRB0 at an input capture trigger due to an external event
- Clearing (to 0) condition
  Writing 1 to ICFCB0 in timer status clear register B (TSCRB)

(9) CMFB0 — Compare Match Flag B0

This status flag indicates occurrence of a match in comparison with output compare register B0 (OCRB0). When 1 is read from this flag, it shows that a match in comparison has occurred in CMFB0. This flag cannot be set to 1 or 0 by the software.

- Setting (to 1) condition
  When the TCNTB0 count value matches the OCRB0 value and the next clock set by the prescaler has occurred
- Clearing (to 0) condition
  Writing 1 to CMFCB0 in the timer status clear register B (TSCRB)

21.5.2.4 TSCRB — Timer Status Clear Register B

Access: 8-bit and 16-bit accessible
Writing 0 is disabled. When 1 is written to this register, data is not retained. This register is always read as 0.

Address: FFE6 0406_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CMFCB6E	CMFCB6M	—	CMFCB12	CMFCB11	CMFCB10	CMFCB6	CMFCB1	ICFCB0	CMFCB0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.30 TSCRB Register Contents

Bit Position	Bit Name	Function
15 to 10	—	These bits are not used. Fix these bits to 0.
9	CMFCB6E	Compare Match Flag Clear B6E Enable 0: Disabled (Initial value) 1: CMFB6E in timer status register B (TSRB) is cleared to 0.
8	CMFCB6M	Compare Match Flag Clear B6M Enable 0: Disabled (Initial value) 1: CMFB6M in timer status register B (TSRB) is cleared to 0.
7	—	This bit is not used. Fix these bits to 0.
6	CMFCB12	Compare Match Flag Clear B12 Enable 0: Disabled (Initial value) 1: CMFB12 in timer status register B (TSRB) is cleared to 0.
5	CMFCB11	Compare Match Flag Clear B11 Enable 0: Disabled (Initial value) 1: CMFB11 in timer status register B (TSRB) is cleared to 0.
4	CMFCB10	Compare Match Flag Clear B10 Enable 0: Disabled (Initial value) 1: CMFB10 in timer status register B (TSRB) is cleared to 0.
3	CMFCB6	Compare Match Flag Clear B6 Enable 0: Disabled (Initial value) 1: CMFB6 in timer status register B (TSRB) is cleared to 0.
2	CMFCB1	Compare Match Flag Clear B1 Enable 0: Disabled (Initial value) 1: CMFB1 in timer status register B (TSRB) is cleared to 0.
1	ICFCB0	Input Capture Flag Clear B0 Enable 0: Disabled (Initial value) 1: ICFB0 in timer status register B (TSRB) is cleared to 0.
0	CMFCB0	Compare Match Flag Clear B0 Enable 0: Disabled (Initial value) 1: CMFB0 in timer status register B (TSRB) is cleared to 0.

The timer status clear register B (TSCRB) is a 16-bit readable/writable register to set clearing of flags at occurrence of an input capture or compare match.

TSCRB can be read and written only in 16-bit units, but this register is always read as 0.

TSCRB is initialized to 0000_H by a reset.

(1) CMFCB6E — Compare Match Flag Clear B6E Enable

Writing 1 to this bit while the compare match flag B6E (CMFB6E) in timer status register B (TSRB) is set to 1 clears CMFB6E to 0. This bit is always read as 0.

(2) CMFCB6M — Compare Match Flag Clear B6M Enable

Writing 1 to this bit while the compare match flag B6M (CMFB6M) in timer status register B (TSRB) is set to 1 clears CMFB6M to 0. This bit is always read as 0.

(3) CMFCB12 — Compare Match Flag Clear B12 Enable

Writing 1 to this bit while the compare match flag B12 (CMFB12) in timer status register B (TSRB) is set to 1 clears CMFB12 to 0. This bit is always read as 0.

(4) CMFCB11 — Compare Match Flag Clear B11 Enable

Writing 1 to this bit while the compare match flag B11 (CMFB11) in timer status register B (TSRB) is set to 1 clears CMFB11 to 0. This bit is always read as 0.

(5) CMFCB10 — Compare Match Flag Clear B10 Enable

Writing 1 to this bit while the compare match flag B10 (CMFB10) in timer status register B (TSRB) is set to 1 clears CMFB10 to 0. This bit is always read as 0.

(6) CMFCB6 — Compare Match Flag Clear B6 Enable

Writing 1 to this bit while the compare match flag B6 (CMFB6) in timer status register B (TSRB) is set to 1 clears CMFB6 to 0. This bit is always read as 0.

(7) CMFCB1 — Compare Match Flag Clear B1 Enable

Writing 1 to this bit while the compare match flag B1 (CMFB1) in timer status register B (TSRB) is set to 1 clears CMFB1 to 0. This bit is always read as 0.

(8) ICFCB0 — Input Capture Flag Clear B0 Enable

Writing 1 to this bit while the input capture flag B0 (ICFB0) in timer status register B (TSRB) is set to 1 clears ICFB0 to 0. This bit is always read as 0.

(9) CMFCB0 — Compare Match Flag Clear B0 Enable

Writing 1 to this bit while the compare match flag B0 (CMFB0) in timer status register B (TSRB) is set to 1 clears CMFB0 to 0. This bit is always read as 0.

21.5.2.5 TICRB — Timer Interrupt Control Register B

Access: 8-bit accessible

Address: FFE6 0408_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	IREGB6E		IREGB6	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 21.31 TICRB Register Contents

Bit Position	Bit Name	Function
7 to 4	—	These bits are not used. Fix these bits to 0.
3, 2	IREGB6E	Interrupt Enable Edge B6E 00: No interrupt request is output. 01: An interrupt request is output when CMFB6 is enabled and then CMFB6M is enabled or when CMFB6M is enabled and then CMFB6 is enabled (AND condition). 10: An interrupt request is output when CMFB6 or CMFB6M is enabled (OR condition) 11: Setting prohibited
1, 0	IREGB6	Interrupt Enable Edge B6 00: An interrupt request is output when CMFB6 is enabled. 01: An interrupt request is output at the next external event input timing after CMFB6 is enabled. 10: An interrupt request is output at the second external event input timing after CMFB6 is enabled. However, no interrupt request is output if compare match B7 occurs before the second external event is input. 11: Setting prohibited

The timer interrupt control register B (TICRB) is an 8-bit readable/writable register to control the compare match interrupt request output timing.

TICRB can be read and written only in 8-bit units.

TICRB is initialized to 00_H by a reset.

(1) IREGB6E — Interrupt Enable Edge B6E

These bits select conditions for CMFB6E interrupt request output of TSRB.

Operation of these bits is disabled if these bits are set to 00.

While these bits are set to 01, an interrupt request is output at the external event input timing at the time when CMFB6M is enabled while a CMFB6 interrupt by the IREGB6 setting is present or at the timing of a CMFB6 interrupt request by the IREGB6 setting with CMFB6M enabled (AND condition). If CMFB6 is cleared even after a CMFB6 interrupt has occurred, no CMFB6E interrupt request is output even while CMFB6M is enabled. Otherwise, if CMFB6M is cleared even while CMFB6M is enabled, a CMFB6 interrupt request by the IREGB6 setting occurs but no CMFB6E interrupt request is output.

While these bits are set to 10, an interrupt request is output when a CMFB6 interrupt by the IREGB6 setting occurs or when CMFB6M is enabled (OR condition). Note that, if the CMFB6 interrupt request timing by the IREGB6 setting differs from the CMFB6M interrupt request timing, respective interrupt requests occur.

However, if the CMFB6 interrupt request timing matches the CMFB6M interrupt request timing, an interrupt request occurs once.

Do not modify these bits while the counter is working or the CMFB6 or CMFB6M bit in TSRB is set to 1. If these bits are modified, this register does not function correctly.

(2) IREGB6 — Interrupt Enable Edge B6

These bits select CMFB6 interrupt request output timing of TSRB. While these bits are set to 00, a CMFB6 interrupt request is output when CMFB6 is enabled. While these bits are set to 01, an interrupt request is output at the first external event input timing after CMFB6 is enabled.

While these bits are set to 10, an interrupt request is output at the second external event input timing after CMFB6 is enabled. However, if compare match B7 occurs during the period of waiting for the second external event input, CMFB6 is automatically cleared and no interrupt request is output even if the second external event is input.

Do not modify these bits while the counter is working or the CMFB6 bit in TSRB is set to 1. If these bits are modified, this register does not function correctly.

21.5.2.6 TCNTB0 — Edge Interval Measuring Counter B0

Access: 32-bit accessible but 8-bit/16-bit in accessible

Address: FFE6 0480_H

Value after reset: $0000\ 0001_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CNTB0															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CNTB0															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.32 TCNTB0 Register Contents

Bit Position	Bit Name	Function
31 to 0	CNTB0	Edge Interval Count These bits store 32-bit counter value

TCNTB0 is a 32-bit readable/writable register that functions as a counter driven by the clock selected in the clock select bit B (CKSELB) of timer control register B (TCRB). TCNTB0 is cleared to $0000\ 0001_H$ on input capture by an externally input event.

TCNTB0 is started when the timer B enable bit (TBE) in ATU-IV master enable register (ATUENR) is set to 1. Clearing the TBE bit to 0 stops the counting but the counter value is not cleared.

TCNTB0 can be read and written only in 32-bit units. Do not make 8-bit or 16-bit accesses to TCNTB0.

TCNTB0 is initialized to $0000\ 0001_H$ by a reset.

21.5.2.7 ICRB0 — Input Capture Register B0

Access: 32-bit accessible but 8-bit/16-bit inaccessible

Address: FFE6 0484_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ICB0															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ICB0															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 21.33 ICRB0 Register Contents

Bit Position	Bit Name	Function
31 to 0	ICB0	Input Capture B0 These bits store 32-bit captured value.

ICRB0 is a 32-bit read-only register that is loaded with the value in TCNTB0 when an externally input event is detected. At this time, bit ICFB0 in timer status register B (TSRB) is set to 1.

An interrupt request can be output to the CPU at this ICRB0 input capture timing. TCNTB0 is cleared to $0000\ 0001_H$.

ICRB0 can be read only in 32-bit units. Do not read ICRB0 in 8-bit or 16-bit units.

ICRB0 is initialized to $0000\ 0000_H$ by a reset.

21.5.2.8 RECRBx— Record Registers Bx (x = 1 to 6)

Access: 32-bit accessible but 8-bit/16-bit inaccessible

Address: FFE6 0488_H (RECRB1)
FFE6 $048C_H$ (RECRB2)
FFE6 0490_H (RECRB3)
FFE6 0494_H (RECRB4)
FFE6 0498_H (RECRB5)
FFE6 $049C_H$ (RECRB6)

Value after reset: 00000000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RECRB1-6															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RECRB1-6															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.34 RECRB1 to RECRB6 Register Contents

Bit Position	Bit Name	Function
31 to 0	RECRB1-6	Record Register B1 to B6 32-bit event capture value

Record registers B1 to B6 (RECRB1 to RECRB6) are 32-bit registers that capture the ICRB0 value at the timing of external event input (event 1 input). Each time an external event is input, the ICRB0 value is captured by shifting the ICRB0 value to RECRB1, the RECRB1 value to RECRB2, and the RECBR2 value to RECRB3. Thus up to seven past TCNT0B values can be retained in ICRB0 and RECRB1 to RECRB6.

RECRB1 to RECRB6 can be read and written only in 32-bit units.

To initialize these registers, write $0000\ 0000_H$ to them.

RECRB1 to RECRB6 are initialized to $0000\ 0000_H$ by a reset.

21.5.2.9 RBURBx — Record Backup Registers Bx (x = 0 to 6)

Access: 32-bit accessible but 8-bit/16-bit inaccessible

Address: FFE6 $04A0_H$ (RBURB0)
FFE6 $04A4_H$ (RBURB1)
FFE6 $04A8_H$ (RBURB2)
FFE6 $04AC_H$ (RBURB3)
FFE6 $04B0_H$ (RBURB4)
FFE6 $04B4_H$ (RBURB5)
FFE6 $04B8_H$ (RBURB6)

Value after reset: 00000000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RBURB0-6															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RBURB0-6															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.35 RBURB0 to RBURB6 Register Contents

Bit Position	Bit Name	Function
31 to 0	RBURB0-6	Record Backup Registers B0 to B6 32-bit event capture backup value

Record backup registers B0 to B6 (RBURB0 to RBURB6) are 32-bit registers that back up values of ICRB0 and RECRB1 to RECRB6 in this register at the timing of event compare match between the event counter B1 (TCNTB1) and OCRB12.

The ICRB0 value is retained in RBURB0, the RECRB1 value in RBURB1, the RECRB2 value in RBURB2, the RECRB3 value in RBURB3, the RECRB4 value in RBURB4, the RECRB5 value in RBURB5, and the RECRB6 value is retained in RBURB6.

RBURB0 to RBURB6 can be read and written only in 32-bit units.

RBURB0 to RBURB6 are initialized to 0000 0000_H by a reset.

21.5.2.10 ICRB3x — Input Capture Registers B3x (x = 0 to 6)

Access: 8-bit accessible

Address: FFE6 $04D8_H$ (ICRB30)
FFE6 $04D9_H$ (ICRB31)
FFE6 $04DA_H$ (ICRB32)
FFE6 $04DB_H$ (ICRB33)
FFE6 $04DC_H$ (ICRB34)
FFE6 $04DD_H$ (ICRB35)
FFE6 $04DE_H$ (ICRB36)

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	ICRB30-36							
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 21.36 ICRB30 to ICRB 36 Register Contents

Bit Position	Bit Name	Function
7 to 0	ICRB30-36	Input Capture Registers B30 to B36 8-bit event capture value

Input capture registers B30 to B36 (ICRB30 to ICRB36) are 8-bit read-only registers. In the edge-interval measuring block, the event counter B1 (TCNTB1) value is captured at occurrence of any of seven external event inputs 1_B to 1_H that are input via timer A (ICRB30 to ICRB36). The event counter B1 (TCNTB1) is not cleared at occurrence of any of external event inputs 1_B to 1_H.

ICRB30 to ICRB36 can be read only in 8-bit units.

ICRB30 to ICRB36 are initialized to 00_H by a reset.

21.5.2.11 OCRB0 — Output Compare Register B0

Access: 32-bit accessible but 8-bit/16-bit inaccessible

Address: FFE6 04BC$_H$

Value after reset: FFFF FFFF$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	OCB0															
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OCB0															
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.37 OCRB0 Register Contents

Bit Position	Bit Name	Function
31 to 0	OCB0	Output Compare B0 These bits store 32-bit data to be compared with TCNTB0.

The output compare register B0 (OCRB0) is a 32-bit readable/writable register that is constantly compared with the input edge interval measuring counter B0 (TCNTB0). When the OCRB0 value matches the TCNTB0 value, an interrupt request due to this compare match can be output to the CPU and the CMFB0 bit in timer status register B (TSRB) is set to 1.

OCRB0 can be read and written only in 32-bit units. Do not make 8-bit or 16-bit accesses to OCRB0.

OCRB0 is initialized to FFFF FFFF$_H$ by a reset.

21.5.2.12 TCNTB1 — Event Counter B1

Access: 8-bit accessible

Address: FFE6 04C4$_H$

Value after reset: 00$_H$

Bit	7	6	5	4	3	2	1	0
	CNTB1							
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.38 TCNTB1 Register Contents

Bit Position	Bit Name	Function
7 to 0	CNTB1	Event Count B1 These bits store 8-bit counter value.

The event counter B1 (TCNTB1) is an 8-bit readable/writable register that counts an external event. According to the setting for the CLRB1 and CLRB1SEL bits in the timer control register B (TCRB), the TCNTB1 value can be cleared to 01$_H$ at the next event input timing after a match in comparison between TCNTB1 and OCRB10 or between TCNTB6 and OCRB6 occurs.

When the TBE bit of the ATU-IV master enable register (ATUENR) is set to 1, the event counter is incremented every time the active level of an external event signal is input.

TCNTB1 can be read and written only in 8-bit units.

TCNTB1 is initialized to 00$_H$ by a reset.

21.5.2.13 OCRB1 — Output Compare Register B1

Access: 8-bit accessible

Address: FFE6 04C6$_H$

Value after reset: FF$_H$

Bit	7	6	5	4	3	2	1	0
	OCB1							
Value after reset	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.39 OCRB1 Register Contents

Bit Position	Bit Name	Function
7 to 0	OCB1	Output Compare B1 These bits store 8-bit data to be compared.

The output compare register B1 (OCRB1) is an 8-bit readable/writable register that is constantly compared with the event counter B1 (TCNTB1). When the OCRB1 value matches the TCNTB1 value, an interrupt request can be output to the CPU and the CMFB1 bit in timer status register B (TSRB) is set to 1 on the next PCLK.

The input capture register B1 (ICRB1) value is transferred to input capture register B2 (ICRB2) and ICRB1 is cleared at this compare match timing.

OCRB1 can be read and written only in 8-bit units.

OCRB1 is initialized to FF$_H$ by a reset.

21.5.2.14 OCRB10 — Output Compare Register B10

Access: 8-bit accessible

Address: FFE6 04C8$_H$

Value after reset: FF$_H$

Bit	7	6	5	4	3	2	1	0
	OCB10							
Value after reset	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.40 OCRB10 Register Contents

Bit Position	Bit Name	Function
7 to 0	OCB10	Output Compare B10 These bits store 8-bit data to be compared.

The output compare register B10 (OCRB10) is an 8-bit readable/writable register that is constantly compared with the event counter B1 (TCNTB1). When the OCRB10 value matches the TCNTB1 value, an interrupt request can be output to the CPU and the CMFB10 bit in timer status register B (TSRB) is set to 1 on the next PCLK.

According to the setting for the CLRB1 and CLRB1SEL bits in timer control register B (TCRB), the TCNTB1 value can be cleared to 01$_H$ at the next event input timing after this compare match occurs. Furthermore, according to the setting for the CLRB3 and CLRB3SEL bits in timer control register B (TCRB), the TCNTB3 value can be cleared to 0000 0000$_H$ at the next PCLK timing after this compare match occurs.

OCRB10 can be read and written only in 8-bit units.

OCRB10 is initialized to FF$_H$ by a reset.

21.5.2.15 OCRB11 — Output Compare Register B11

Access: 8-bit accessible

Address: FFE6 04CA$_H$

Value after reset: FF$_H$

Bit	7	6	5	4	3	2	1	0
	OCB11							
Value after reset	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.41 OCRB11 Register Contents

Bit Position	Bit Name	Function
7 to 0	OCB11	Output Compare B11 These bits store 8-bit data to be compared.

The output compare register B11 (OCRB11) is an 8-bit readable/writable register. According to the setting for the PIMRSEL bit in timer control register B (TCRB), this register can operate by using PIMR2 for down-count values to be loaded to RLDB and for down-count values of TCNTB2 during a compare match between TCNTB1 and OCRB11. PIMR1 is always used for up-count values of TCNTB3.

When the TCNTB1 value matches the OCRB11 value, an interrupt request due to this compare match can be output to the CPU and the CMFB11 bit in timer status register B (TSRB) is set to 1 on the next PCLK.

OCRB11 can be read and written only in 8-bit units.

OCRB11 is initialized to FF$_H$ by a reset.

21.5.2.16 OCRB12 — Output Compare Register B12

Access: 8-bit accessible

Address: FFE6 04CC$_H$

Value after reset: FF$_H$

Bit	7	6	5	4	3	2	1	0
	OCB12							
Value after reset	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.42 OCRB12 Register Contents

Bit Position	Bit Name	Function
7 to 0	OCB12	Output Compare B12 These bits store 8-bit data to be compared.

The output compare register B12 (OCRB12) is an 8-bit readable/writable register. Values of ICRB0 and RECRB1 to RECRB6 can be backed up in RBURB0 to RBURB6 at the timing of event compare match between the event counter B1 (TCNTB1) and this register.

When the TCNTB1 value matches the OCRB12 value, an interrupt request due to this compare match can be output to the CPU and the CMFB12 bit in timer status register B (TSRB) is set to 1 on the next PCLK.

OCRB12 can be read and written only in 8-bit units.

OCRB12 is initialized to FF$_H$ by a reset.

21.5.2.17 ICRB1 — Input Capture Register B1

Access: 32-bit accessible but 8-bit/16-bit inaccessible

Address: FFE6 04D0$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ICB1															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ICB1															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 21.43 ICRB1 Register Contents

Bit Position	Bit Name	Function
31 to 0	ICB1	Input Capture B1 These bits store 32-bit input capture value.

ICRB1 is a 32-bit read-only register. ICRB1 accumulates and stores the input inter-edge measuring counter B0 (TCNTB0) value in ICRB1 at the input timing of the external event. ICRB1 is cleared on compare match between the event counter B1 (TCNTB1) and output compare register B1 (OCRB1).

ICRB1 can be read only in 32-bit units. Do not read ICRB1 in 8-bit or 16-bit units.

ICRB1 is initialized to 0000 0000$_H$ by a reset.

21.5.2.18 ICRB2 — Input Capture Register B2

Access: 32-bit accessible but 8-bit/16-bit inaccessible

Address: FFE6 04D4$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ICB2															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ICB2															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 21.44 ICRB2 Register Contents

Bit Position	Bit Name	Function
31 to 0	ICB2	Input Capture B2 These bits store 32-bit input capture value.

ICRB2 is a 32-bit read-only register. ICRB2 latches the value in input capture register B1 (ICRB1) at the time of a match in comparison between event counter B1 (TCNTB1) and output compare register B1 (OCRB1).

ICRB2 can be read only in 32-bit units. Do not read ICRB2 in 16-bit units.

ICRB2 is initialized to 0000 0000$_H$ by a reset.

21.5.2.19 LDB — Load Register B

Access: 32-bit accessible but 8-bit/16-bit inaccessible

Address: FFE6 0500_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	LDVAL							
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LDVAL															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.45 LDB Register Contents

Bit Position	Bit Name	Function
31 to 24	—	These bits are not used. Fix these bits to 0.
23 to 0	LDVAL	Load Value These bits store 24-bit data to be loaded to TCNTB2 and RLDB.

LDB is a 32-bit readable/writable register that is mapped to the lower-order 24 bits on a 32-bit boundary. The lower 24 bits are available.

When the LDSEL bit in timer I/O control register B (TIORB) is set to 1, the value in this register is loaded to reload counter B2 (TCNTB2) and reload register B (RLDB).

LDB can be read and written only in 32-bit units. Do not make 16-bit accesses to LDB.

LDB is initialized to $0000\ 0000_H$ by a reset.

21.5.2.20 RLDB — Reload Register B

Access: 32-bit accessible but 8-bit/16-bit inaccessible

Address: FFE6 0504$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	RLDVAL															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	RLDVAL								—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Table 21.46 RLDB Register Contents

Bit Position	Bit Name	Function
31 to 8	RLDVAL	Reload Value These bits store 24-bit reload value.
7 to 0	—	These bits are not used. Fix these bits to 0.

RLDB is a 32-bit readable/writable register that is aligned with a longword boundary. The upper 24 bits are available.

This register is updated when the externally input event is detected while the LDEN bit in timer I/O control register B (TIORB) is cleared to 0.

The value in input capture register B0 (ICRB0) or load register B (LDB) minus the value in the PIMR bits in PIM is used for updating. According to the setting for the PIMRSEL bit in timer control register B (TCRB) and a compare match between TCNTB1 and OCRB11, PIMR2 is used as the PIMR value. In other cases, PIMR1 is used as the PIMR value. ICRB0 or LDB is set by the LDSEL bit in TIORB. For subtraction on ICRB0 and PIMR, the lower 24 bits of ICRB0 and PIM which is zero-extended.

The value in this register is added to the value in TCNTB2 on the first counter clock after the value in bits reload count B2 (CNTB2) is equal to or less than the value in the pulse interval multiplier register (PIM).

RLDB can be read and written only in 32-bit units. Do not make 8-bit or 16-bit accesses to RLDB.

RLDB is initialized to 0000 0000$_H$ by a reset.

An initial value should be set before starting this function. For details on the operation, see **Section 21.5.3.2, Frequency-Multiplied Clock Generator**.

21.5.2.21 TCNTB2 — Reload Counter B2

Access: 32-bit accessible but 8-bit/16-bit inaccessible

Address: FFE6 0508_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CNTB2															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CNTB2								—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Table 21.47 TCNTB2 Register Contents

Bit Position	Bit Name	Function
31 to 8	CNTB2	Reload Counter B2 These bits store 24-bit reload counter value.
7 to 0	—	These bits are not used. Fix these bits to 0.

TCNTB2 is a down counter mapped to 24 bits in a 32-bit readable and writable register and functions as a counter driven by the clock selected by the clock selection B bits (CKSELB0 to CKSELB2) in timer control register B (TCRB). Each decrementation by the value set in the pulse interval multiplier register (PIMR). According to the setting for the PIMRSEL bit in timer control register B (TCRB) and in response to a compare match between TCNTB1 and OCRB11, PIMR2 is used as the PIMR value. In other cases, PIMR1 is used as the PIMR value.

TCNTB2 starts counting when the TBE bit in the ATU-IV master enable register (ATUENR) is set to 1. Even if the TBE bit is cleared to 0, this counter is not cleared.

The TCNTB2 value is updated when the externally input event is detected while the LEDN bit in timer I/O control register B (TIORB) is cleared to 0.

The value in input capture register B0 (ICRB0) or load register B (LDB) is used for updating.

ICRB0 or LDB can be set by the LDSEL bit in TIORB.

The value in reload register B (RLDB) is added to the value stored in this counter on the first counter clock cycle after the value in this counter is equal to or less than the value in PIM. The frequency-multiplied clock (AGCK1) is output as a train of single pulses equal in width to the cycle of the Pϕ clock on reloading.

TCNTB2 can be read and written only in 32-bit units. Do not make 8-bit or 16-bit accesses to TCNTB2.

TCNTB2 is initialized to $0000\ 0000_H$ by a reset.

An initial value should be set before starting this function. For details on the operation, see **Section 21.5.3.2, Frequency-Multiplied Clock Generator**.

21.5.2.22 PIMR1 — Pulse Interval Multiplier Register 1

Access: 16-bit accessible but 8-bit inaccessible

Address: FFE6 050C$_H$

Value after reset: 0001$_H$

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	PIM1											
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.48 PIMR1 Register Contents

Bit Position	Bit Name	Function
15 to 12	—	These bits are not used. Fix these bits to 0.
11 to 0	PIM1	Pulse Interval Multiplier 1 These bits set the multiplication ratio for the frequency-multiplied clock. The settable value ranges from 1 to 4095.

PIMR1 is a 16-bit readable/writable register that sets the multiplication ratio of the externally input event for generation of the frequency-multiplied clock.

The settable value ranges from 1 (001$_H$) to 4095 (FFF$_H$). Do not set the PIM bits to 000$_H$. If the PIM bits are set to 000$_H$ operation cannot be guaranteed.

The value in this register is used in various registers: as the step size in decrementation of reload counter B2 (TCNTB2); for calculation of the value to be input to reload register B (RLDB); and for calculation of the value to be input to corrected event counter B3 (TCNTB3). However, when a match in comparison between TCNTB1 and OCRB11 occurs with the PIMRSEL bit in timer control register B (TCRB) set to 1, PIMR2 is used to calculate the TCNTB2 down-count value and the RLDB input value. PIMR1 is always used to calculate the TCNTB3 input value.

PIMR1 can be read and written only in 16-bit units. Do not make 8-bit accesses to PIMR1.

PIMR1 is initialized to 0001$_H$ by a reset.

21.5.2.23 PIMR2 — Pulse Interval Multiplier Register 2

Access: 16-bit accessible but 8-bit inaccessible

Address: FFE6 050E$_H$

Value after reset: 0001$_H$

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	PIM2											
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.49 PIMR2 Register Contents

Bit Position	Bit Name	Function
15 to 12	—	These bits are not used. Fix these bits to 0.
11 to 0	PIM2	Pulse Interval Multiplier 2 These bits set the multiplication ratio for the frequency-multiplied clock. The settable value ranges from 1 to 4095.

The pulse interval multiplier register 2 (PIMR2) is a 16-bit readable/writable register that is mapped to the lower 12 bits. PIMR2 sets the multiplication ratio of the frequency-multiplied clock when tooth is missing.

The settable value ranges from 1 (001$_H$) to 4095 (FFF$_H$). Do not set the PIM2 bits to 000$_H$. If the PIM2 bits are set to 000$_H$, operation cannot be guaranteed.

By setting the PIMRSEL bit in timer control register B (TCRB), the reload counter B2 (TCNTB2) performs down-counting with the set PIMR2 value (PIM2) (not with the PIMR1 value) while the TCNTB1 value is equal to the OCRB11 value. The PIMR2 value is also used to calculate the reload register B (RLDB) input value, but is not used to calculate the correction event counter B3 (TCNTB3) input value. The multiplication ratio of the frequency-multiplied clock at missing tooth is corrected by setting a value of (N (number of missing teeth) + 1) times for the PIMR1 value. When the number of missing teeth is 2, for example, a value of (PIMR1 value × 3) is set for PIMR2.

PIMR2 can be read and written only in 16-bit units. Do not make 8-bit accesses to PIMR2.

PIMR2 is initialized to 0001$_H$ by a reset.

21.5.2.24 TCNTB6 — Multiplied Clock Counter B6

Access: 32-bit accessible but 8-bit/16-bit inaccessible

Address: FFE6 0510_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CNTB6															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CNTB6				—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R

Table 21.50 TCNTB6 Register Contents

Bit Position	Bit Name	Function
31 to 12	CNTB6	Frequency-Multiplied Clock Counter B6 These bits store 20-bit counter value driven by the frequency-multiplied clock
11 to 0	—	These bits are not used. Fix these bits to 0.

The multiplied clock counter B6 (TCNTB6) is a 20-bit up-counter mapped in the 32-bit readable/writable register. TCNTB6 performs up-counting on the frequency-multiplied clock (AGCK1). This counter is cleared to 00000_H when the external input event is detected.

TCNTB6 starts counting when the TBE bit in the ATU-IV master enable register (ATUENR) is set to 1. Even if the TBE bit is cleared to 0 and counting up is stopped, the counter value is not cleared.

TCNTB6 can be read and written only in 32-bit units. Do not make 16-bit accesses to TCNTB6.

TCNTB6 is initialized to $0000\ 0000_H$ by a reset.

21.5.2.25 ICRB6 — Input Capture Register B6

Access: 32-bit accessible but 8-bit/16-bit inaccessible

Address: FFE6 0514_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ICB6															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ICB6				—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 21.51 ICRB6 Register Contents

Bit Position	Bit Name	Function
31 to 12	ICB6	Input Capture B6 20-bit input capture value
11 to 0	—	These bits are not used. Fix these bits to 0.

The input capture register B6 (ICRB6) is a 32-bit read-only register that is mapped to upper 20 bits. No value can be written to this register.

When an external event input is detected, ICRB6 stores the TCNTB6 value at the next clock (Pϕ) timing. The value stored in ICRB6 is compared with the TCNTB6M value at the next external event input timing. When the ICRB6 value is larger than the TCNTB6M value, an interrupt request can be output.

ICRB6 can be read only in 32-bit units.

ICRB6 is initialized to $0000\ 0000_H$ by a reset.

21.5.2.26 RARB6 — Multiplication Setting Register B6

Access: 8-bit accessible

Address: FFE6 0518$_H$

Value after reset: 40$_H$

Bit	7	6	5	4	3	2	1	0
	RARB6							
Value after reset	0	1	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.52 RARB6 Register Contents

Bit Position	Bit Name	Function
7 to 0	RARB6	Multiplication Setting Register B6 These bits set the TCNTB6M count value.

The multiplication setting register B6 (RARB6) is an 8-bit readable/writable register to set an up-count value of the frequency-multiplied clock counter B6M (TCNTB6M). RARB6 handles the lower 6 bits as fixed point. A multiplication of 1 to 3.984375 can be set with a resolution of 1/64.

RARB6 can be read and written only in 8-bit units.

RARB6 is initialized to 40$_H$ by a reset.

21.5.2.27 TCNTB6M — Frequency-Multiplied Clock Counter B6M

Access: 32-bit accessible but 8-bit/16-bit inaccessible

Address: FFE6 051C$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CNTB6M															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CNTB6M										—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R

Table 21.53 TCNTB6M Register Contents

Bit Position	Bit Name	Function
31 to 6	CNTB6M	Frequency-Multiplied Clock Counter B6M 26-bit frequency-multiplied clock count value
5 to 0	—	These bits are not used. Fix these bits to 0.

The frequency-multiplied clock counter B6M (TCNTB6M) is a 32-bit readable/writable register (up-counter) that is mapped to upper 26 bits. TCNTB6M performs count-up operation with a count value of multiplication that is set in multiplication setting register B6 (RARB6) for the frequency-multiplied clock (AGCK1). TCNTB6M handles the lower 6 bits (b11 to b6) as fixed point, and can count within a range of 1 to 1048575.984375 with a resolution of 1/64. Each time an external event is input, the TCNTB6M value is compared with the ICRB6 value. When ICRB6 is larger than TCNTB6M, an interrupt request can be output. TCNTB6M is initialized to 0000 0000$_H$ at the first clock (Pϕ) timing after occurrence of an external event input.

Setting the TBE bit in the ATU-IV master enable register (ATUENR) to 1 starts counting by this counter. Clearing the TBE bit to 0 stops counting but the counter value is not cleared.

TCNTB6M can be read and written only in 32-bit units. Do not make 8-bit or 16-bit accesses to TCNTB6M.

TCNTB6M is initialized to 0000 0000$_H$ by a reset.

21.5.2.28 OCRB6 — Output Compare Register B6

Access: 32-bit accessible but 8-bit/16-bit inaccessible

Address: FFE6 0520_H

Value after reset: FFFF $F000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	OCB6															
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OCB6				—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R

Table 21.54 OCRB6 Register Contents

Bit Position	Bit Name	Function
31 to 12	OCB6	Output Compare B6 These bits store 20-bit data to be compared.
11 to 0	—	These bits are not used. Fix these bits to 0.

OCRB6 is a 32-bit readable/writable register in which output compare B6 (OCB6) is mapped to upper 20 bits in 32 bits. Whether to perform compare match between multiplied-clock counter B6 and OCRB6 can be selected by the IOB6 setting of timer I/O control register B (TIORB).

The OCRB6 value is compared with the TCNTB6 value. When input of the frequency-multiplied clock signal (AGCK1) leads to the values matching, a missing tooth detection trigger is generated, which leads to setting of the CMFB6 bit in timer status register B (TSRB) to 1 and can also cause the output of an interrupt request for the CPU.

According to the setting for the CLRB1 and CLRB1SEL bits in the timer control register B (TCRB) and this compare match occurrence, the TCNTB1 value can be cleared to 01_H at the next event input timing. Furthermore, according to the setting for the CLRB3 and CLRB3SEL bits in the timer control register B (TCRB) and this compare match occurrence, the TCNTB3 value can be cleared to $0000\ 0000_H$ at the next PCLK timing.

OCRB6 can be read and written only in 32-bit units. Do not make 8-bit or 16-bit accesses to OCRB6.

OCRB6 is initialized to FFFF $F000_H$ by a reset.

21.5.2.29 OCRB7 — Output Compare Register B7

Access: 32-bit accessible but 8-bit/16-bit inaccessible

Address: FFE6 0524_H

Value after reset: FFFF $F000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	OCB7															
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OCB7				—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R

Table 21.55 OCRB7 Register Contents

Bit Position	Bit Name	Function
31 to 12	OCB7	Output Compare B7 These bit store 20-bit data to be compared.
11 to 0	—	These bits are not used. Fix these bits to 0.

OCRB7 is a 32-bit readable/writable register in which output compare B7 (OCB7) is mapped to upper 20 bits in 32 bits.

The OCRB7 value is compared with the TCNTB6 value. When the frequency-multiplied clock (AGCK1) is input with these values matching, a compare match occurs. When the IREGB6 bits in timer interrupt control register B (TICRB) are set to 10, the CMFB6 bit in timer status register B (TSRB) is cleared to 0 at an occurrence of this compare match.

There is no status flag or interrupt request that indicates occurrence of a compare match of OCRB7.

OCRB7 can be read and written only in 32-bit units. Do not make 16-bit accesses to OCRB7.

OCRB7 is initialized to FFFF $F000_H$ by a reset.

21.5.2.30 TCNTB3 — Correcting Event Counter B3

Access: 32-bit accessible but 8-bit/16-bit inaccessible

Address: FFE6 0580$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CNTB3															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CNTB3				—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R

Table 21.56 TCNTB3 Register Contents

Bit Position	Bit Name	Function
31 to 12	CNTB3	Correcting Event Counter B3 These bits store 20-bit event count value.
11 to 0	—	These bits are not used. Fix these bits to 0.

The correcting event counter B3 (TCNTB3) is a 20-bit up-counter mapped in the 32-bit readable/writable register.

TCNTB3 transfers its counter value to the correcting multiplied-clock counter B4 (TCNTB4) at the external event input timing, and then starts up-counting. The pulse interval multiplier register 1 (PIMR1) value is used for up-counting (CNTB3+ = PIM1). The pulse interval multiplier register 2 (PIMR2) value is not used for up-counting of TCNTB3.

TCNTB3 starts counting when the TBE bit in the ATU-IV master enable register (ATUENR) is set to 1. Even if the TBE bit is cleared to 0, the counter value is not cleared. According to the settings for the CLRB3 and CLRB3SEL bits in the timer control register B (TCRB), the TCNTB3 value can be cleared to 0000 0000$_H$ at the first PCLK timing after a compare match between TCNTB1 and OCRB10, TCNTB6 and OCRB6, or TCNTB3 and OCRB8.

TCNTB3 can be read and written only in 32-bit units. Do not make 8-bit or 16-bit accesses to TCNTB3.

TCNTB3 is initialized to 0000 0000$_H$ by a reset.

21.5.2.31 OCRB8 — Output Compare Register B8

Access: 32-bit accessible but 8-bit/16-bit inaccessible

Address: FFE6 0594$_H$

Value after reset: FFFF F000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	OCB8															
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OCB8				—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R

Table 21.57 OCRB8 Register Contents

Bit Position	Bit Name	Function
31 to 12	OCB8	Output Compare B8 20-bit comparison value
11 to 0	—	These bits are not used. Fix these bits to 0.

OCRB8 is a 32-bit readable/writable register in which output compare B8 (OCB8) bits are mapped to the upper 20 bits.

The OCRB8 value is compared with the TCNTB3 value. When TCNTB3 clear (due to a compare match between OCRB8 and TCNTB3) is enabled by bit 4 and bit 3 in the timer control register B (TCRB), the TCNTB3 value is cleared to 0 at the first PCLK timing after a compare match between the OCRB8 value and the TCNTB3 value. OCRB8 has no compare match status flag or interrupt.

OCRB8 can be read and written only in 32-bit units. Do not read or write it in 16-bit units.

OCRB8 is initialized to FFFF F000$_H$ by a reset.

21.5.2.32 TCNTB4 — Multiplied-and-Corrected Clock Counter B4

Access: 32-bit accessible but 8-bit/16-bit inaccessible

Address: FFE6 0584_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CNTB4															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CNTB4				—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R

Table 21.58 TCNTB4 Register Contents

Bit Position	Bit Name	Function
31 to 12	CNTB4	Multiplied-and-Corrected Clock Counter B4 These bits store 20-bit frequency-multiplied clock count value.
11 to 0	—	These bits are not used. Fix these bits to 0.

The multiplied-and-corrected clock counter B4 (TCNTB4) is a 20-bit up-counter mapped in the 32-bit readable/writable register.

The value in TCNTB3 is loaded to this counter when the externally input event is detected.

This up-counter is driven by the frequency-multiplied clock (AGCK1) output by reload counter B2 (TCNTB2) and is cleared to 00000_H when the externally input event is detected while CNTB3 = 00000_H.

When the values in this counter and TCNTB3 match with the CCS bit in timer I/O control register B (TIORB) set to 1, counting is stopped. When the CCS bit is set to 0, counting is continued.

TCNTB4 starts counting when the TBE bit in the ATU-IV master enable register (ATUENR) is set to 1. Even if the TBE bit is cleared to 0 and counting is stopped, the counter value is not cleared.

TCNTB4 can be read and written only in 32-bit units. Do not make 16-bit accesses to TCNTB4.

TCNTB4 is initialized to $0000\ 0000_H$ by a reset.

21.5.2.33 TCNTB5 — Multiplied-and-Corrected Clock Generating Counter B5

Access: 32-bit accessible but 8-bit/16-bit inaccessible

Address: FFE6 0588$_H$

Value after reset: 0000 1000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CNTB5															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CNTB5				—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R

Table 21.59 TCNTB5 Register Contents

Bit Position	Bit Name	Function
31 to 12	CNTB5	Multiplied-and-Corrected Clock Generation Count B5 These bit store 20-bit multiplied-and-corrected clock count value.
11 to 0	—	These bits are not used. Fix these bits to 0.

The multiplied-and-corrected clock generating counter B5 (TCNTB5) is a 20-bit up-counter mapped in the 32-bit readable/writable register.

This counter is enabled by the TBE bit in ATU-IV master enable register (ATUENR) and the count control B5 bit (CTCNTB5). Incrementation of this counter is driven by the PCLK as long as the counter value is less than the value in multiplied-and-corrected clock counter B4 (TCNTB4). Incrementation is stopped when the counter value matches the value in correcting counter clearing register (TCCLRB).

The value in this counter is corrected and cleared when the externally input event is detected while CNTB3 = 00000$_H$ or bit TCCLFSB in the correcting counter clearing flag set register B (TCCLFSRB) is set to 1. The value for the clearing depends on the counter value.

- When the counter value is equal to the value in TCCLRB
  This counter is cleared to 0000 1000$_H$. The corrected counter clear flag register B (TCCLFRB) retains 0.
- When the counter value is not equal to the value in TCCLRB
  Incrementation continues until the counter value reaches the value in TCCLRB and then TCNTB5 is cleared to 0000 1000$_H$.

Every incrementation of this counter, a single pulse of the multiplied-and-corrected clock (AGCKM) is output. The clock can be output on clock-bus line 5 by setting the CB5SEL bit in clock bus control register (CBCNT) to 1. The output of the clock is temporarily stopped by altering the setting in bit CTCNTB5 in the TIORB during counting.

TCNTB5 can be read and written only in 32-bit units. Do not make 16-bit accesses to TCNTB5.

TCNTB5 is initialized to 0000 1000$_H$ by a reset.

An initial value should be set before starting this function. For details on the operation, see **Section 21.5.3.3, Frequency-Multiplied Clock Signal Corrector**.

21.5.2.34 TCCLFRB — Correcting Counter Clear Flag Register B

Access: 8-bit accessible
Address: FFE6 058C$_H$
Value after reset: 00$_H$

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TCCLFB
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 21.60 TCCLFRB Register Contents

Bit Position	Bit Name	Function
7 to 1	—	These bits are not used. Fix these bits to 0.
0	TCCLFB	Correcting Counter Clear Flag B 0: TCNTB5 is not automatically counting up (Initial value). 1: TCNTB5 is automatically counting up until it matches the TCCLRB value.

The correcting counter clear flag register B (TCCLFRB) is an 8-bit read-only register that indicates whether the function of correcting TCNTB5 (the automatic up-counting function that continues counting until TCNTB5 = TCCLRB) is enabled or disabled.

TCCLFRB can be read only in 8-bit units.

TCCLFRB is initialized to 00$_H$ by a reset.

(1) TCCLFB — Correcting Counter Clear Flag B

This status flag shows that the function of correcting TCNTB5 is enabled and the automatic up-counting function continues until TCNTB5 = TCCLRB. When 1 is read from this flag, it shows that TCNTB5 is automatically up-counting until TCNTB5 = TCCLRB. This flag cannot be set to 1 or 0 by the software.

- Setting (to 1) conditions
  - Writing 1 to TCCLFSB in the correcting counter clear flag setting register B (TCCLFSRB)
  - When an external event is input with TCNTB3 = 00000$_H$ and TCNTB5 ≠ TCCLRB while TCNTB5 is automatically up-counting until it reaches the TCCLRB value
- Clearing (to 0) conditions
  - Writing 1 to TCCLFCB in the correcting counter clear flag clearing register B (TCCLFCRB)
  - When TCNTB5 value has reached the TCCLRB value after counting with the TCCLFB bit set to 1, the TCNTB5 value has matched the TCCLRB value, and TCNTB5 has been cleared to 0000 1000$_H$

21.5.2.35 TCCLFSRB — Correcting Counter Clear Flag Setting Register B

Access: 8-bit accessible
Writing 0 is disabled. If 1 is written to this register, data is not retained. This register is always read as 0.

Address: FFE6 058D$_H$

Value after reset: 00$_H$

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TCCLFSB
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 21.61 TCCLFSRB Register Contents

Bit Position	Bit Name	Function
7 to 1	—	These bits are not used. Fix these bits to 0.
0	TCCLFSB	Correcting Counter Clear Flag Set B 0: Disabled 1: TCCLFB in correcting counter clear flag register B (TCCLFRB) is set to 1.

The correcting counter clear flag setting register B (TCCLFSRB) is an 8-bit readable/writable register.

TCCLFSRB can be read and written only in 8-bit units, but it is always read as 0.

TCCLFSRB is initialized to 00$_H$ by a reset.

(1) TCCLFSB — Correcting Counter Clear Flag Set B

Writing 1 to TCCLFSB sets TCCLFB in the correcting counter clear flag register B (TCCLFRB) to 1 to enable automatic up-counting until the TCNTB5 matches the TCCLRB value. When TCCLFSB is set to 1 while TCCLFB is 1, TCCLFB holds 1. When TCCLFSB is set to 1 while TCNTB5 = TCCLRB, TCNTB5 is cleared to 0000 1000$_H$ and TCCLFB holds 0. When writing 0 to this flag is attempted, the writing is ignored.

21.5.2.36 TCCLFCRB — Correcting Counter Clear Flag Clearing Register B

Access: 8-bit accessible
Writing 0 is disabled. If 1 is written to this register, data is not retained. This register is always read as 0.

Address: FFE6 058E$_H$

Value after reset: 00$_H$

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TCCLFCB
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 21.62 TCCLFCRB Register Contents

Bit Position	Bit Name	Function
7 to 1	—	These bits are not used. Fix these bits to 0.
0	TCCLFCB	Correcting Counter Clear Flag Clear B 0: Disabled 1: TCCLFB in correcting counter clear flag register B (TCCLFRB) is cleared to 0.

The correcting counter clear flag clearing register B (TCCLFCRB) is an 8-bit readable/writable register.

TCCLFCRB can be read and written only in 8-bit units, but it is always read as 0.

TCCLFCRB is initialized to 00$_H$ by a reset.

(1) TCCLFCB — Correcting Counter Clear Flag Clear B

Writing 1 to TCCLFCB sets TCCLFB in the correcting counter clear flag register B (TCCLFRB) to 0 to inhibit automatic up-counting until the TCNTB5 matches the TCCLRB value and enter the up-count state of TCNTB5 at the time when TCNTB4 > TCNTB5. If this conflicts with the condition for setting the TCCLFB bit when TCNTB3 = 0 and an event input has occurred, clearing by writing this register is prioritized. If TCCLFB is cleared when it is 0, TCCLFB holds 0. When writing 0 to this flag is attempted, the writing is ignored.

21.5.2.37 TCCLRB — Correcting Counter Clearing Register B

Access: 32-bit accessible but 8-bit/16-bit inaccessible

Address: FFE6 0590_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CCLRB															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CCLRB				—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R

Table 21.63 TCCLRB Register Contents

Bit Position	Bit Name	Function
31 to 12	CCLRB	Correcting Counter Clear B These bits store 20-bit correcting counter clear value.
11 to 0	—	These bits are not used. Fix these bits to 0.

The correcting counter clearing register B (TCCLRB) is a 32-bit readable/writable register mapped to upper 20 bits in 32 bits.

TCCLRB is constantly compared with TCNTB5. When they match, TCNTB5 is stopped and a counter clearing trigger is output to timer D. TCNT1Dx and TCNT2Dx in timer D are separately cleared by setting the corresponding counter clearing enable bit in timer control register Dx (TCRDx).

TCCLRB can be read and written only in 32-bit units. Do not make 8/16-bit accesses to TCCLRB.

TCCLRB is initialized to $0000\ 0000_H$ by a reset.

21.5.2.38 ACRTRGB — AGCKM2 Correction Enable Setting Register

Access: 8-bit accessible
Writing 0 is disabled. If 1 is written to these bits, data is not retained. These bits are always read as 0.

Address: FFE6 05A0$_H$

Value after reset: 00$_H$

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ACRTRG
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 21.64 ACRTRGB Register Contents

Bit Position	Bit Name	Function
7 to 1	—	These bits are not used. Fix these bits to 0.
0	ACRTRG	AGCKM2 Correction Enable 0: Disabled 1: AGCKM2 correction is enabled.

The AGCKM2 correction enable setting register (ACRTRGB) is an 8-bit readable/writable register.

When the ACRTRG bit is set to 1, the number of correction clocks is corrected according to the specification of the ACRVALRB register.

Then the AGCKM2 output corrected based on AGCKM from the input timer B external event (AGCK) is generated.

Write-access to the AGCKM2 correction enable setting register (ACRTRGB) is enabled only when ACRST is 00$_B$.

When ACRST is not 00$_B$, write-access to this register is disabled.

ACRTRGB can be read and written only in 8-bit units.

ACRTRGB is initialized to 00$_H$ by a reset.

21.5.2.39 ACRCLRB — AGCKM2 Correction Clear Setting Register

Access: 8-bit accessible
Writing 0 is disabled. If 1 is written to these bits, data is not retained. These bits are always read as 0.

Address: FFE6 05A1$_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ACRCLR
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 21.65 ACRCLRB Register Contents

Bit Position	Bit Name	Function
7 to 1	—	These bits are not used. Fix these bits to 0.
0	ACRCLR	AGCKM2 Correction Clear 0: Disabled 1: AGCKM2 correction is stopped. ACRSTRB = 00_B

The AGCKM2 correction clear setting register (ACRCLRB) is an 8-bit readable/writable register.

Setting the ACRCLR bit to 1 transitions ACRSTRB to State 00_B and stops AGCKM correction control.

ACRTCLR can be read and written only in 8-bit units.

ACRTCLR is initialized to 00_H by a reset.

21.5.2.40 ACRSTRB — AGCKM2 Correction Status Register

Access: 8-bit accessible

Address: FFE6 05A2$_H$

Value after reset: 00$_H$

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	ACRST	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 21.66 ACRSTRB Register Contents

Bit Position	Bit Name	Function
7 to 2	—	These bits are not used. Fix these bits to 0.
1, 0	ACRST	AGCKM2 Correction Status 00: No AGCKM2 correction 01: Correction waiting state The ACREN bit is set to 1, and then an external event input is waited. 10: AGCKM2 correction in progress

The AGCKM2 correction status register (ACRSTRB) is an 8-bit readable register.

The ACRST bit is used to check the AGCKM1 correction status.

ACRSTRB can be read only in 8-bit units.

ACRSTRB is initialized to 00$_H$ by a reset.

21.5.2.41 ACRVALRB — AGCKM2 Correction Clock Count Setting Register

Access: 16-bit accessible but 8-bit/32-bit inaccessible

Address: FFE6 05A4$_H$

Value after reset: 0000$_H$

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ACRSIGN	ACRVAL[14:0]														
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.67 ACRVALRB Register Contents

Bit Position	Bit Name	Function
15	ACRSIGN	AGCKM2 Correction Clock Count Sign 0: The number of correction clocks (ACRVAL[14:0] value) is treated as a positive value. 1: The number of correction clocks (ACRVAL[14:0] value) is treated as a negative value.
14 to 0	ACRVAL[14:0]	AGCKM2 Correction Clock Count These bits specify the number of AGCKM2 correction clocks. AGCKM2 is corrected for the number of clocks indicated by the ACRSIGN bit and these bits. (Positive value: high level is output additionally, Negative value: AGCKM high level is masked to low level and is output as AGCKM2.)

The AGCKM2 correction clock count setting register (ACRVALRB) is a 16 bit readable/writable register.

This register generates AGCKM2 output corrected based on AGCKM according to the set value of the AGCKM2 correction clock count setting register (ACRVALRB).

The AGCKM2 correction counter starts counting from the set ACRVALB value and continues down-counting until 0001$_H$ is detected.

When ACRVALB is set to 0000$_H$, the counter decrements as 0000$_H$ → 7FFF$_H$ → 7FFE$_H$ → 7FFD$_H$ → … → 0001$_H$.

ACRVALRB can be read and written only in 16-bit units.

ACRVALRB is initialized to 0000$_H$ by a reset.

21.5.3 Detailed Operation

21.5.3.1 Edge Interval Measuring Function and Edge Input Stopping Function

For timer B, input-capture and compare-match operations are unconditionally performed with 32-bit input capture register B0 (ICRB0) and 32-bit output compare register B0 (OCRB0), respectively. These registers are connected to input edge-interval measuring counter B0 (TCNTB0).

Operation of timer B is started by setting the TBE bit in the ATU-IV master enable register (ATUENR).

ICRB0 captures the TCNTB0 value when an event (the AGCK signal) is input via timer A. At the same time, TCNTB0 is set to 0000 0001_H. An interrupt request can be output to the CPU at this capture timing. This enables measurement of the time of edge-to-edge interval.

The value captured in ICRB0 is transferred to the frequency-multiplied clock generating block to be used as an input value to the reload counter B2 (TCNTB2) and the reload register B (RLDB). Furthermore, up to six captured values (ICRB0) can be retained in the record registers B1 to B6 (RECRB1 to RECRB6) at the external input event timing. Values retained in ICRB0 and RECRB1 to RECRB6 are backed up in the record backup registers B0 to B6 (RBURB0 to RBURB6) at the timing of B12 (OCRB12) event compare match between TCNTB1 and the output compare register B12 (OCRB12).

When TCNTB0 reaches the value set in OCRB0, an interrupt request due to this compare match can be output. This interrupt indicates that active edge input has stopped for at least time equivalent to the setting in OCRB0.

Figure 21.16 shows input-capture and compare-match operation of TCNTB0.

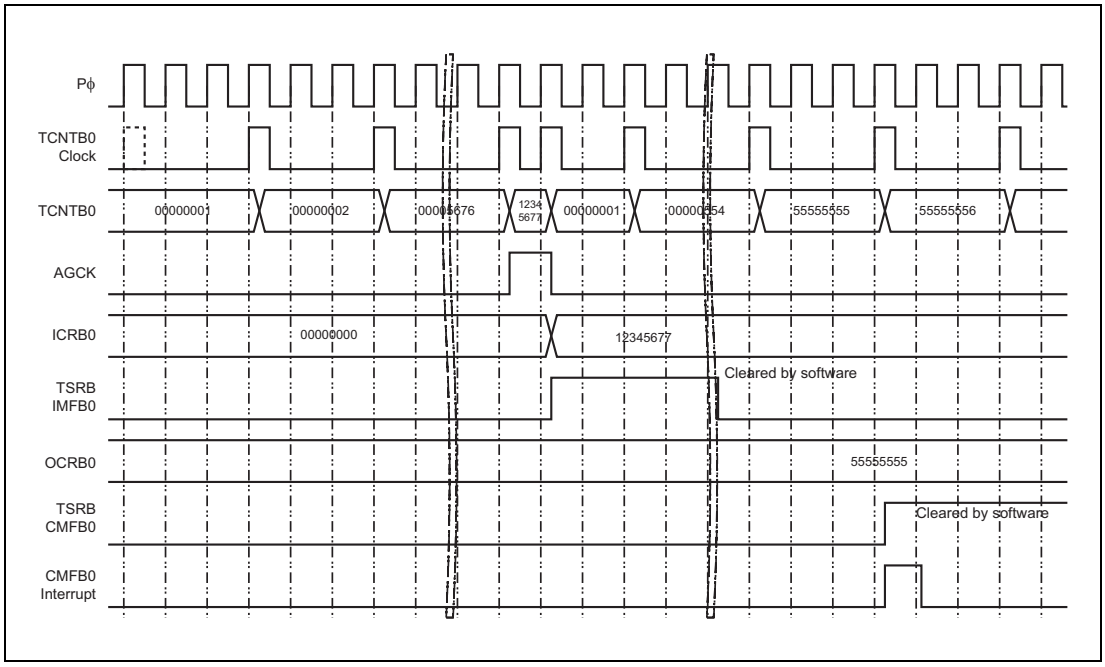


Figure 21.16 Input-Capture and Compare-Match Operation of TCNTB0

Event counter B1 (TCNTB1) counts the input of signals indicating external events (AGCK). When a predetermined value is set in output compare register B1 (OCRB1) and TCNTB1 reaches that value, a compare-match occurs. When a value is set in output compare register B10 (OCRB10) by the setting of the TCNTB1 clear setting bit (CLRB1) in the timer control register B (TCRB), TCNTB1 can be cleared at the first event input timing after an event compare match between TCNTB1 and OCRB10. Furthermore, by the setting of the TCNTB3 clear setting bit (CLRB3SEL) in the timer control register

B (TCRB), TCNTB3 can be cleared at a compare match between TCNTB1 and OCRB10. Input capture register B2 (ICRB2) can capture input capture register B1 (ICRB1) at the timing of a compare match between TCNTB1 and OCRB1. This compare match makes it possible to output an interrupt request to the CPU, allowing detection of edge input stop of external event. In addition, event counter B1 (TCNTB1) has a function for compare match with output compare register B11 (OCRB11). When the pulse interval multiplier select bit (PIMRSELR) is set to 1, TCNTB1 can operate by using PIMR2 for down-count values to be loaded to RLDB and for down-count values of TCNTB2 at a compare match between TCNTB1 and OCRB11. When the TCNTB1 value matches the OCRB11 value, an interrupt request due to this compare match can be output to the CPU.

The external event signal (AGCK) drives the capturing of TCNTB0 values in ICRB1. Moreover, latching of ICRB1 in ICRB2 on matches between TCNTB1 and OCRB1 can also be selected. This enables the measurement of multiple edge-to-edge intervals.

Registers ICRB30 to ICRB 36 capture the TCNTB1 value by using the external event input 1B to 1H as a trigger. ICRB30 corresponds to external event input 1B and ICRB31 to ICRB36 correspond to respective external event input 1C to 1H.

Figure 21.17 shows compare-match operation of TCNTB1 and capture operation of ICRB1 and ICRB2, and **Figure 21.18** shows capture operation of TCNTB1 and ICRB3x.

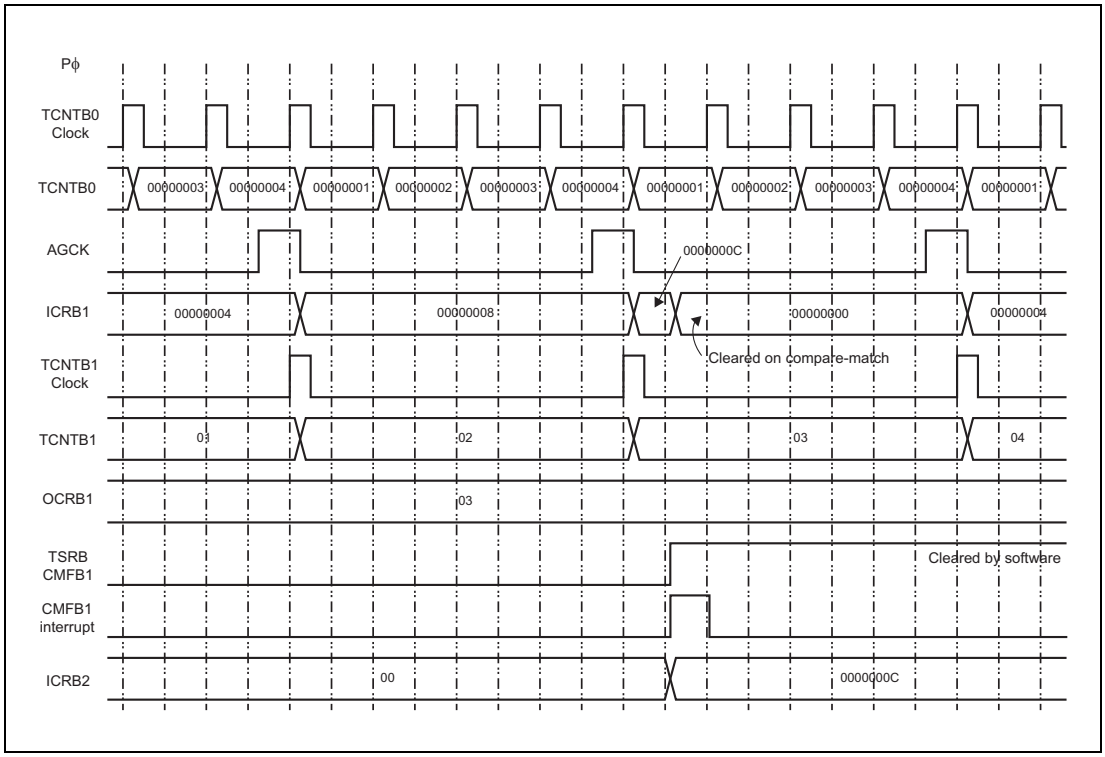


Figure 21.17 Compare-Match Operation of TCNTB1 and Capture Operation of ICRB1 and ICRB2

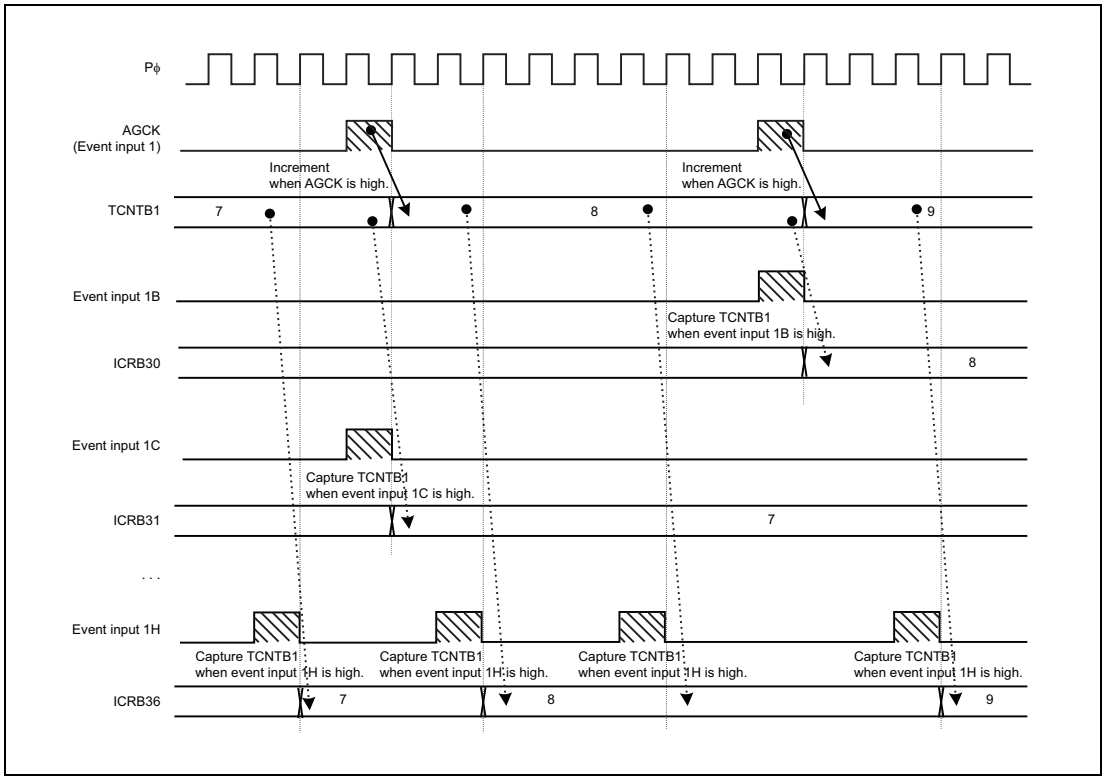


Figure 21.18 Count Operation of TCNTB1 and Capture Operation of ICRB3x

21.5.3.2 Frequency-Multiplied Clock Generator

The frequency-multiplied clock generator generates a clock signal (AGCK1) for use within timer B. The cycle of this clock signal is obtained by dividing the intervals between selected transitions of the external-event input (AGCK) by the value in the pulse interval multiplier register (PIMR1 or PIMR2).

When the LDEN bit in the TIORB is 0, On the selected transition of the external-event input signal, the lower 24 bits of the value captured in ICRB0 of the edge-interval measuring block are transferred to 24-bit reload counter B2 (TCNTB2). At the same time, the value transferred to TCNTB2 minus the value in PIMR (PIM1 or PIM2) is transferred to 24-bit reload register B (RLDB). When the LDEN bit in the TIORB is 1, the values of TCNTB2 and RLDB are not updated on ICRB0 input capture.

When the LDSEL bit in timer I/O control register B (TIORB) is set to 1, the value in load register B (LDB) instead of that in ICRB0 can be transferred to TCNTB2 and RLDB. When the pulse interval multiplier select bit (PIMRSELR) is set to 1, the above operation is performed using the PIMR2 value instead of PIMR1 value while TCNTB1 matches OCRB11.

24-bit reload counter B2 (TCNTB2) is driven to count down by the clock selected by the CKSELB bits in timer control register B (TCRB). Each decrementation is by the value set in PIMR (TCNTB2- = PIM). When the value in the down counter is less than or equal to that of the PIM bits, RLDB is automatically read out into TCNTB2 and added to the count value (TCNTB2 + the RLDB value) and , which again starts to count down. A single pulse of the multiplied clock signal (AGCK1) is output in synchronization with the reloading of TCNTB2. The pulse width is equal to the cycle of the PCLK.

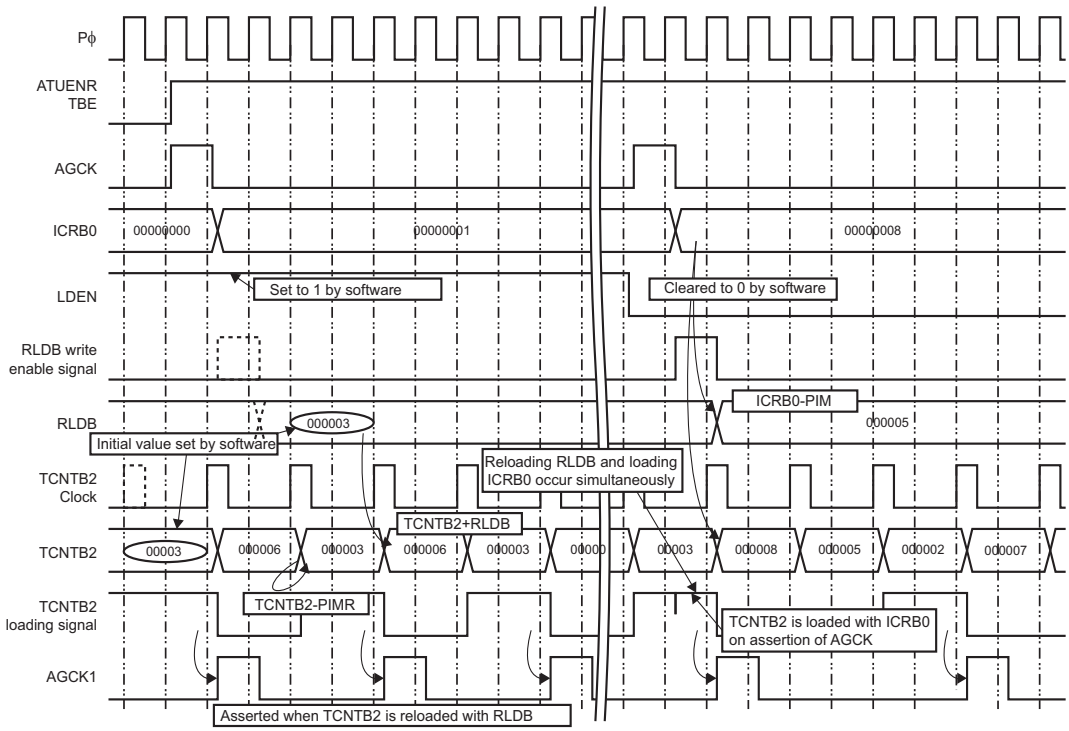
Compare match operation by frequency-multipiled clock (AGCK1) can be performed by using 20-bit output compare register B6 (OCRB6) and 20-bit frequency-multiplied clock counter B6 (TCNTB6).

Incrementaion of TCNTB6 on the assertion of AGCK1 is unconditional. The values in TCNTB6 and output compare register B6 (OCRB6) are tested for matches, and an interrupt request will be generated to the CPU when the values match. The IREGB6 bits in TICRB can be set so that the interrupt is generated on the match, on the first AGCK pulse (external event) after the match, or on the second AGCK pulse (external event) after the match.

When an external event input (AGCK) is detected, ICRB6 stores the TCNTB6 value at the next clock (Pϕ) timing. The frequency-multiplied clock counter B6M (TCNTB6M) performs count-up operation with a count value of multiplication that is set in multiplication setting register B6 (RARB6) for the frequency-multiplied clock (AGCK1). Each time an external event (AGCK) is input, the TCNTB6M value is compared with the ICRB6 value. When the ICRB6 value is larger than the TCNTB6M value as a result of comparison, an interrupt request can be output.

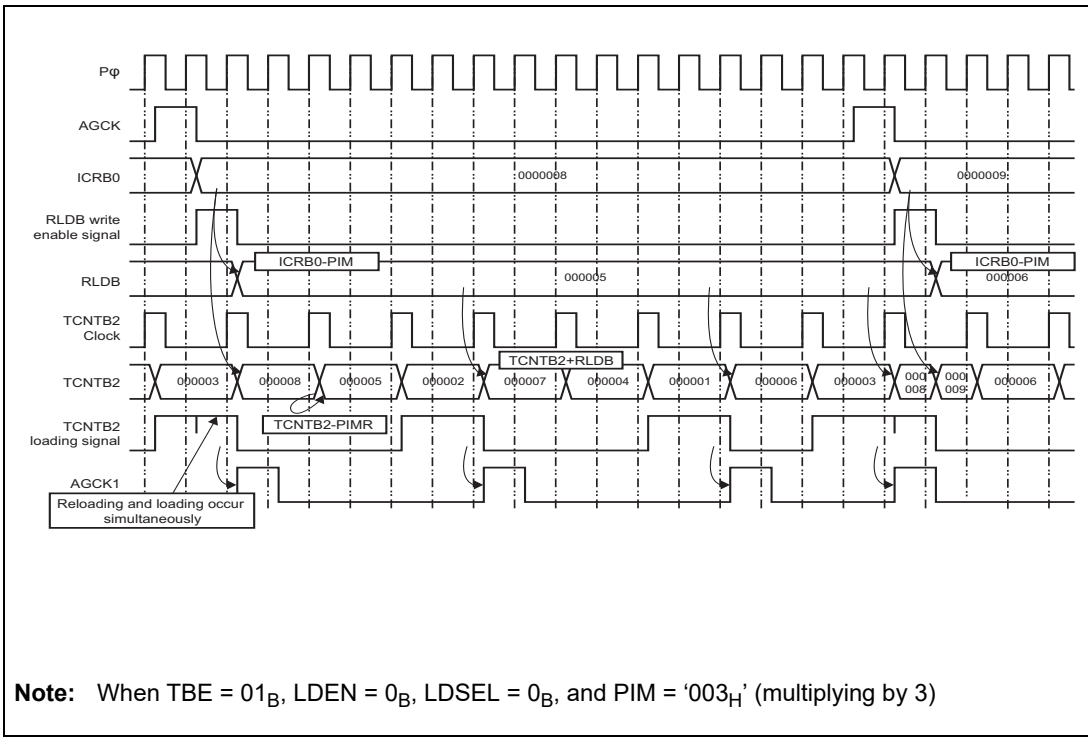
Since AGCK1 is generated with reference to the previous edge-to-edge interval, if two consecutive edge intervals differ significantly, the clock will not be generated correctly. To correct this error, AGCK1 is corrected by the frequency-multiplied clock signal corrector, whose description is provided in the next section, and is to be converted into AGCKM.

Figure 21.19 and **Figure 21.20** show counting operations with reloading and output of the frequency-multiplied clock and **Figure 21.21** and **Figure 21.22** show the generation of interrupt requests on matches between TCNTB6 and CMFB6, enabled or disabled by the setting in IREG. **Figure 21.23** shows the Switching Timing of PIMR1 and PIMR2. **Figure 21.24**. shows comparison of TCNTB6M and ICRB6 operations and interrupt request output.



Note: When LDSEL = 0_B and PIM = '003_H' (multiplying by 3)

Figure 21.19 Counting Operations with Reloading and Output of Frequency-Multiplied Clock (1)



Note: When TBE = 01_B, LDEN = 0_B, LDSEL = 0_B, and PIM = '003_H' (multiplying by 3)

Figure 21.20 Counting Operations with Reloading and Output of Frequency-Multiplied Clock (2)

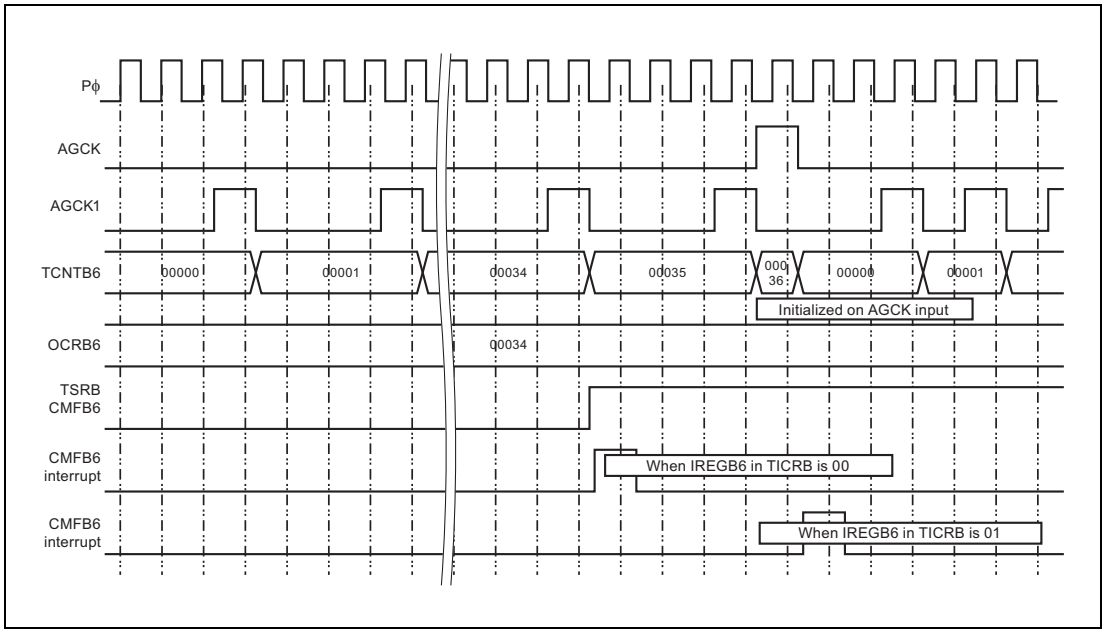
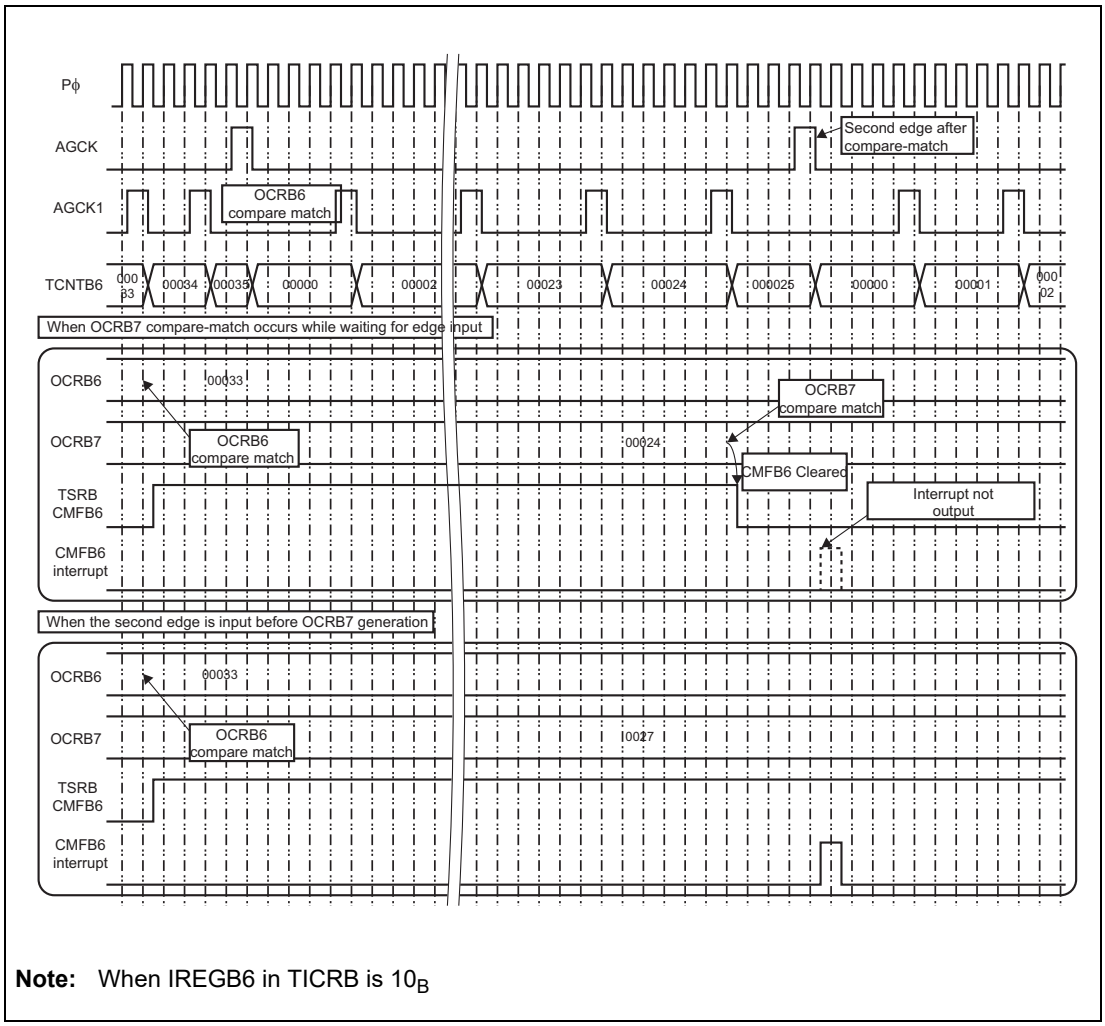


Figure 21.21 Compare-Match Operation of TCNTB6 and Output of CMFB6 Interrupt (IREGB6 = 00, 01)



Note: When IREGB6 in TICRB is 10_B

Figure 21.22 CMFB6 Interrupt Output when IREGB6 = 10

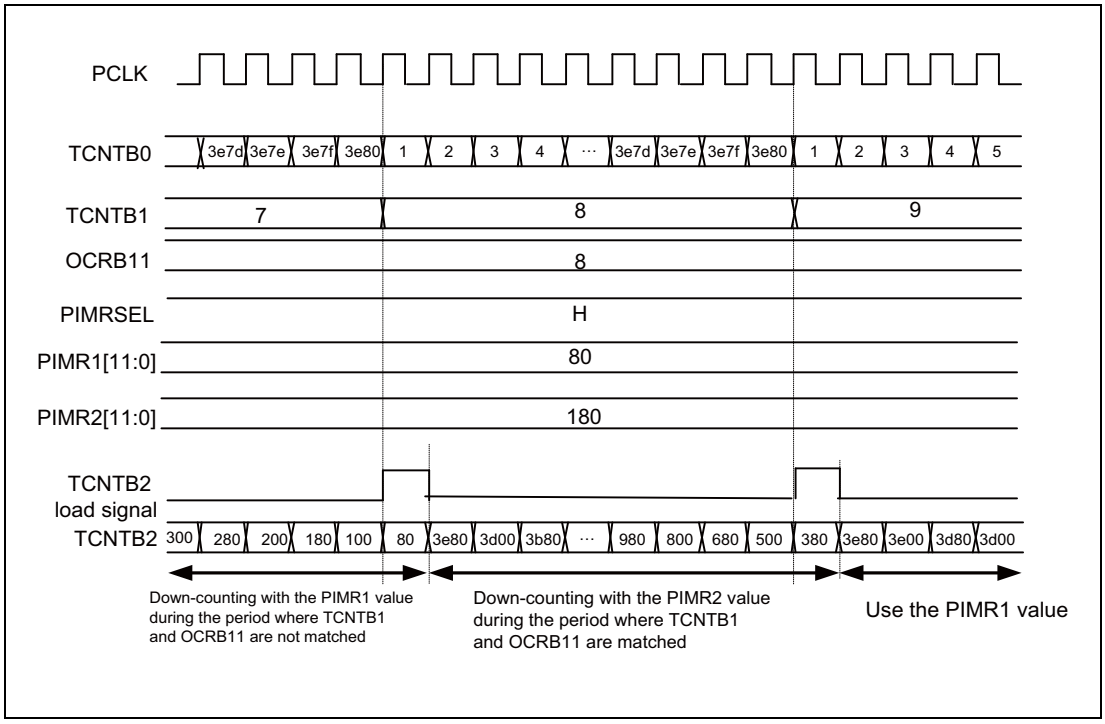


Figure 21.23 Switch Timing between PIMR1 and PIMR2

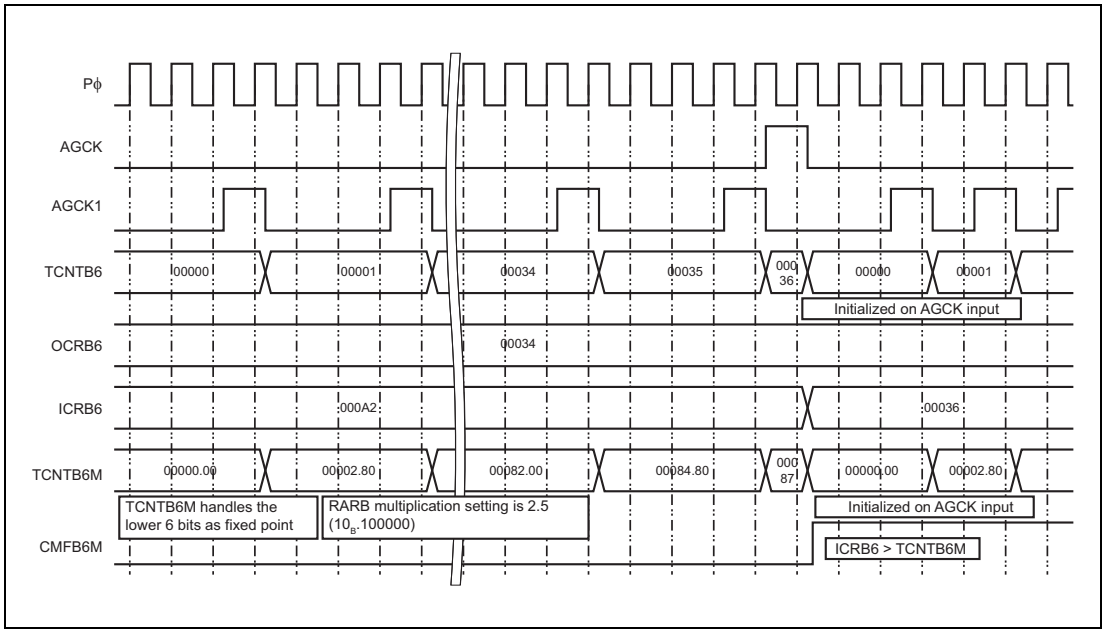


Figure 21.24 Example of Operations of TCNTB6M and ICRB6

21.5.3.3 Frequency-Multiplied Clock Signal Corrector

The frequency-multiplied clock signal which is generated by dividing the intervals between external event inputs by the multiplication ratio set in the PIM bits in PIMR can be corrected by using 20-bit correcting event counter B3 (TCNTB3), 20-bit multiplied-and-corrected clock counter B4 (TCNTB4), 20-bit multiplied-and-corrected clock generating counter B5 (TCNTB5), and correcting counter clearing register B (TCCLRB).

TCNTB3 is a 20-bit up-counter that is driven by the external event input (AGCK). On the selected transition of the AGCK signal, the value in TCNTB3 is transferred to TCNTB4, after which TCNTB3 is incremented by the value in the PIM bits (TCNTB3+ = PIM). According to the setting for the CLRB3SEL bit (TCNTB3 clear select bit) in the timer control register B (TCRB), TCNTB3 can be cleared by a compare match between TCNTB1 and OCRB10, between TCNTB6 and OCRB6, or between TCNTB3 and OCRB8.

TCNTB4 is a 20-bit up-counter that is driven by the multiplied clock signal (AGCK1). TCNTB3 is loaded to TCNTB4 with the AGCK input as a trigger, and incrementation of TCNTB4 is driven by the AGCK1 input.

The counter correcting select bit (CCS) in TIORB controls counting by TCNTB4; that is, it selects whether or not counting stops when TCNTB3 = TCNTB4.

TCNTB5 is a 20-bit up-counter that is driven by the PCLK clock, meaning that it operates at a high speed. TCNTB5 is constantly compared with TCNTB4 and is incremented as long as its value is lower than that in TCNTB4. Each time TCNTB5 is actually incremented, it produces a single pulse whose width is equal to one cycle of the PCLK clock. pulse of the peripheral clock signal, namely the multiplied-and -corrected clock signal (AGCKM), for which output on clock-bus line 5 can be selected by bit CB5SEL in the clock bus control register (CBCNT). The AGCKM signal is then available on clock-bus line 5 as a source to drive counting by other timers.

As state above, TCNTB5 is not incremented when its value is greater that in TCNTB4 (for example, after TCNTB3 has been loaded to TCNTB4), TCNTB5 can also be disabled by the count control B5 (CTCNTB5) bit in timer I/O control register B (TIORB). This halts the output of the AGCKM signal.

As long as its value is lower than TCNTB4, TCNTB5 is incremented until it reaches the value in correcting counter clearing register B (TCCLRB). This function allows you to check the automatic up-count status by using correcting counter clear flag register B (TCCLFRB). It also allows you to set automatic up-count operation status by using the correcting counter clear setting register (TCCLFSRB) and the correcting counter clear flag clearing register (TCCLFCRB). Incrementation of TCNTB5 then stops, regardless of the relation between its value and that of TCNTB4. In addition, timer counters 1Dx and 2Dx of timer D (TCNT1Dx and TCNT2Dx) can be separately cleared by this match as a trigger when the corresponding counter clearing enable bit (C1CEDx/C2CEDx) in timer control register Dx (TCRDx) is set to 1.

TCNTB4 is unconditionally cleared to 0000 0000_H when a pulse of the external-event signal (AGCK) is input while TCNTB3 = 0000 0000_H. TCNTB5 is unconditionally set to 0000 1000_H when a pulse of the external-event signal (AGCK) is input while TCNTB3 = 0000 0000_H.

However, when TCNTB5 has not reached TCCLRB, TCNTB5 is incremented until it reaches TCCLRB. After that, it is set to 0000 1000_H.

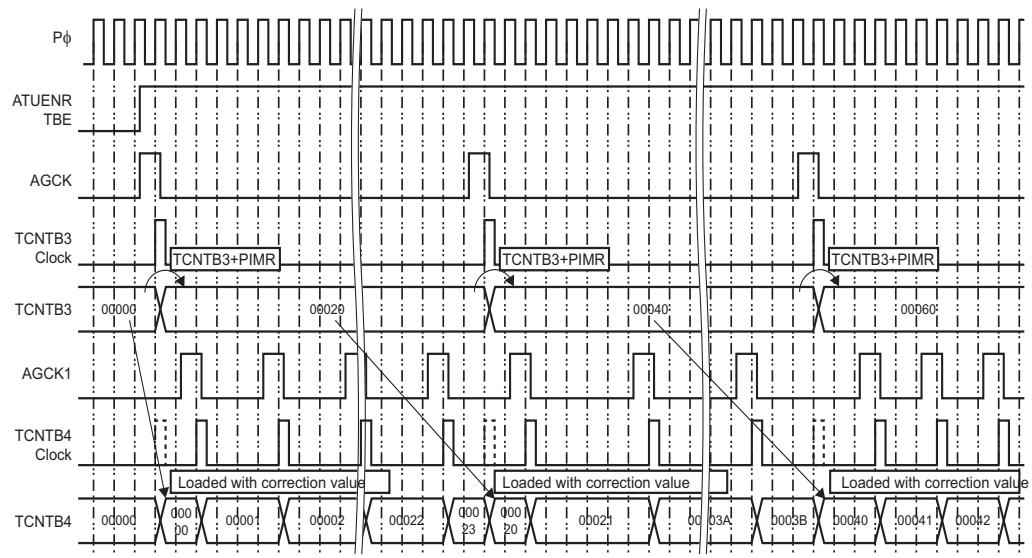
Figure 21.25 shows operations of TCNTB3 and TCNTB4, **Figure 21.26** shows operation when TCNTB5 is being started up, **Figure 21.27** shows TCNTB5 operation with correction at the end of a cycle, and **Figure 21.28** shows TCNTB5 operation with no correction at the end of a cycle.

The multiplied-and-corrected clock 2 (AGCKM2) is generated and output for the multiplied-and-corrected clock (AGCKM) by setting the correction enable trigger bit (ACRTRG) to 1. In the case of

positive logic (+) correction for AGCKM, AGCKM2 extends the active time of AGCKM. In the case of negative logic (–) correction, AGCKM2 masks the active time of AGCKM.

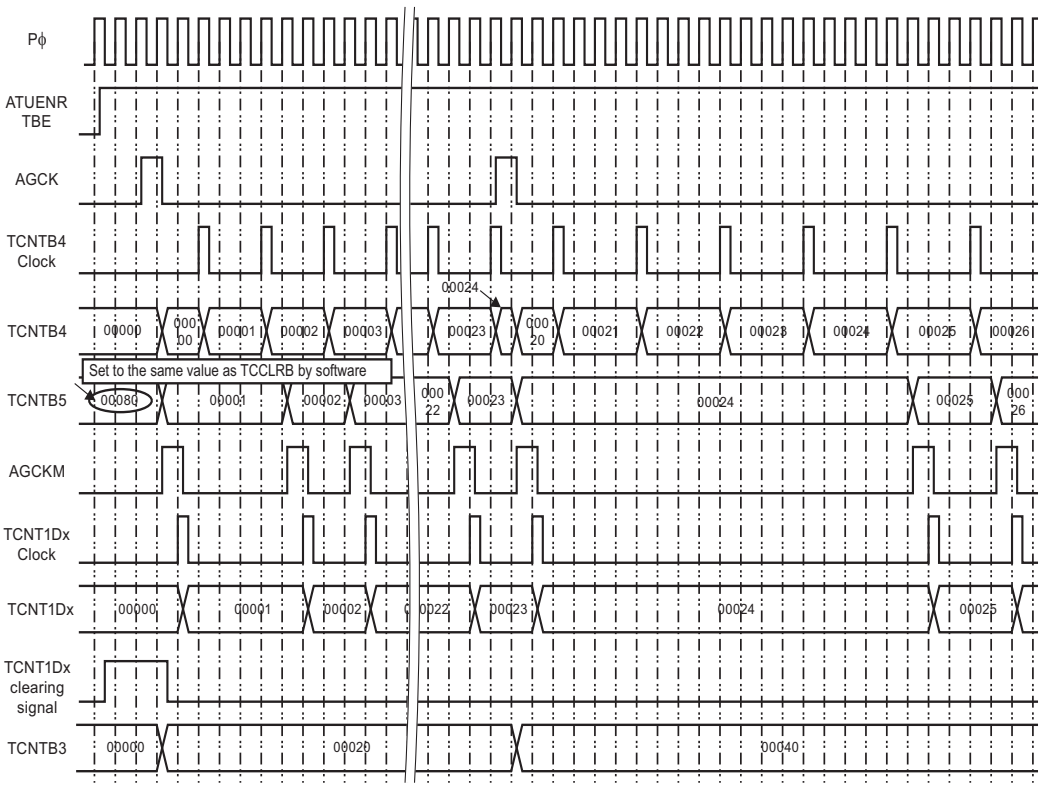
Generation of AGCKM2 is controlled by ACRSTRB, and the correction time is controlled by the internal down-counter.

AGCKM2 is generated as follows: AGCKM2 correction is started (ACRSTRB = 10_B) by detection of a timer B external event (AGCK) after correction enable trigger bit (ACRTRG) is set to 1 (ACRSTRB = 01_B). **Figure 21.29** shows AGCKM2 Operation (Positive Logic) and **Figure 21.30** shows AGCKM2 Operation (Negative Logic).



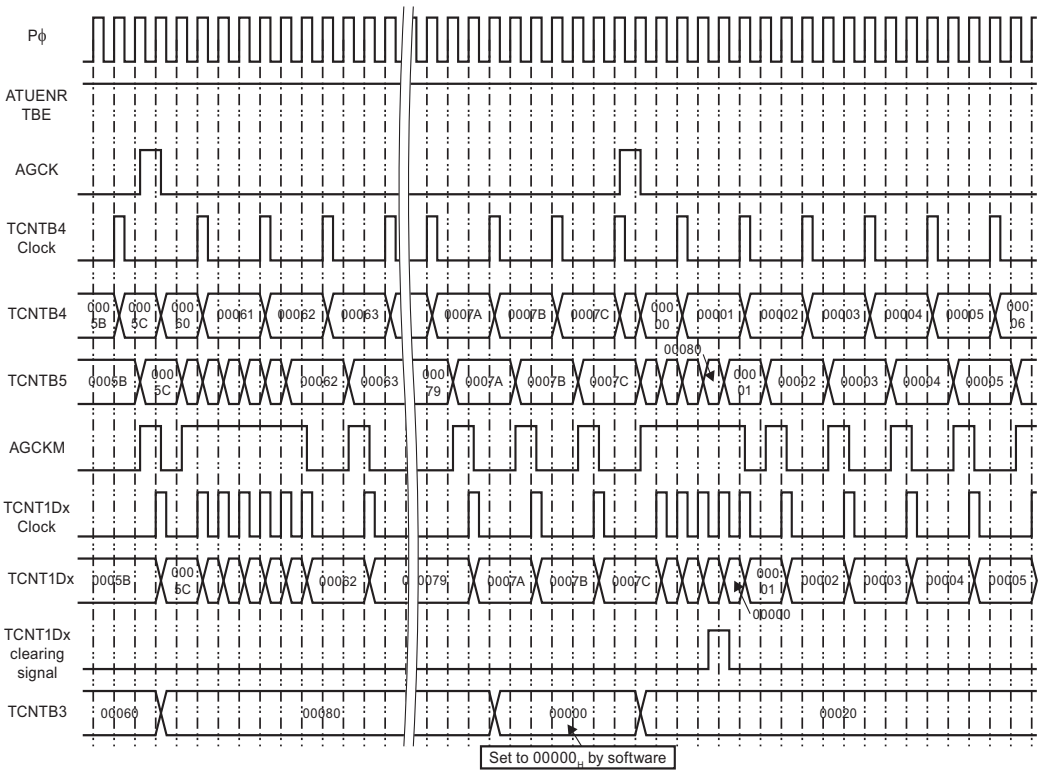
Note: When PIM = '020_H' (multiplying by 32)

Figure 21.25 Operation of TCNTB3 and TCNTB4



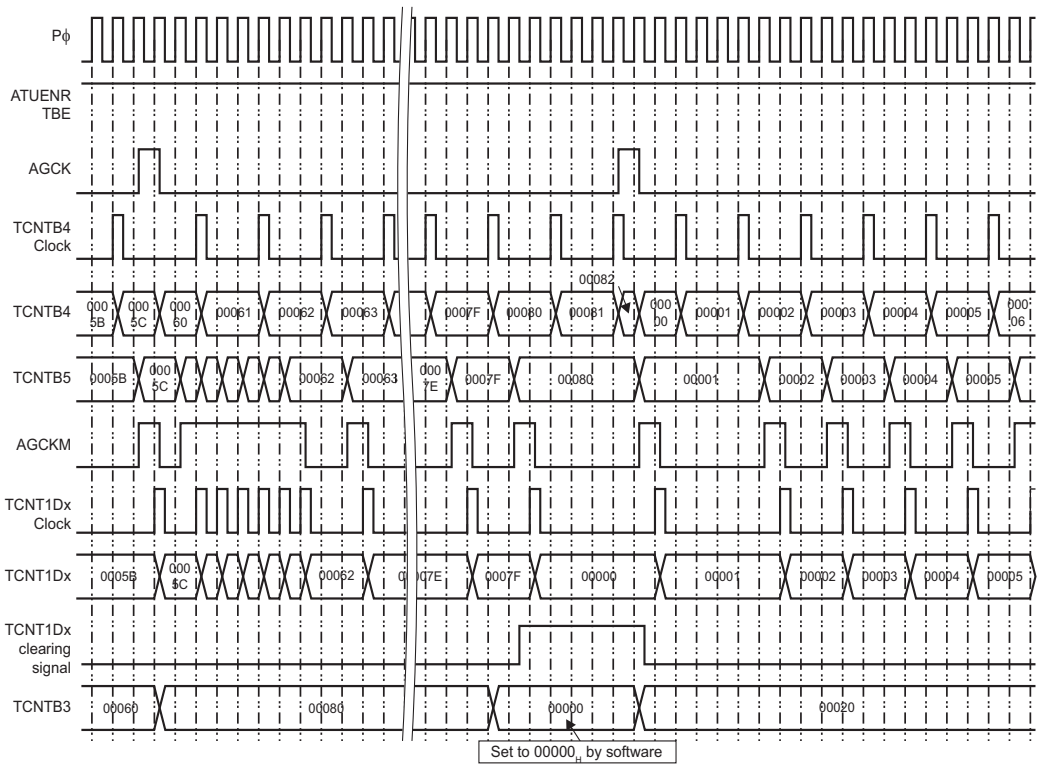
Note: When PIM = '020_H' (multiplying by 32) and TCCLRB = 00080_H
This figure shows the case of TCNT1Dx. This is the same for TCNT2Dx.

Figure 21.26 TCNTB5 Operation (at Start-Up)



Note: When PIM = '020_H' (multiplying by 32) and TCCLRB = 00080_H
This figure shows the case of TCNT1Dx. This is the same for TCNT2Dx.

Figure 21.27 Operation of TCNTB5 (with Correction at End of Cycle)



Note: When PIM = '020_H' (multiplying by 32) and TCCLRB = 00080_H
This figure shows the case of TCNT1Dx. This is the same for TCNT2Dx.

Figure 21.28 Operation of TCNTB5 (without Correction at End of Cycle)

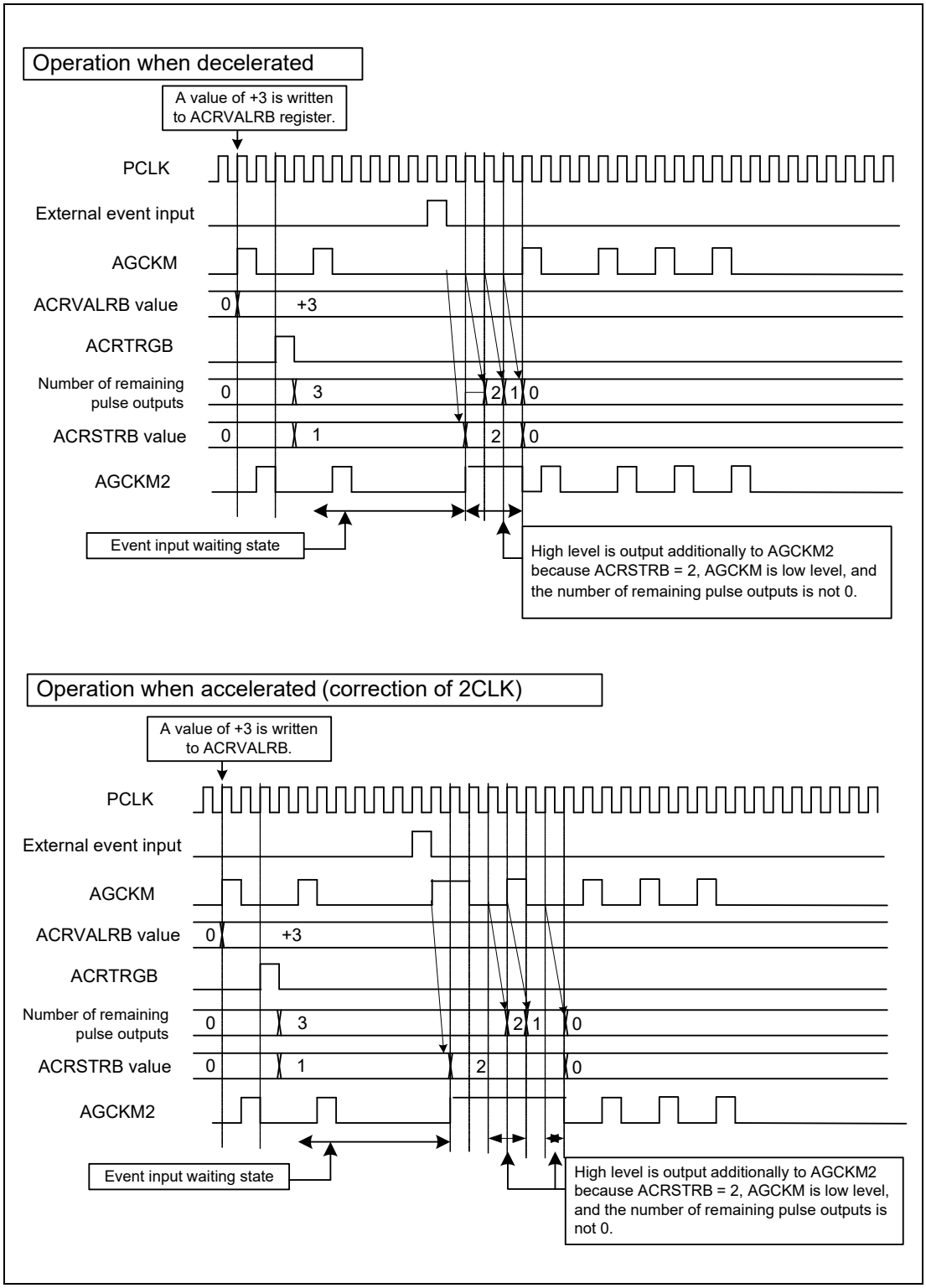


Figure 21.29 AGCKM2 Operation (Positive Logic)

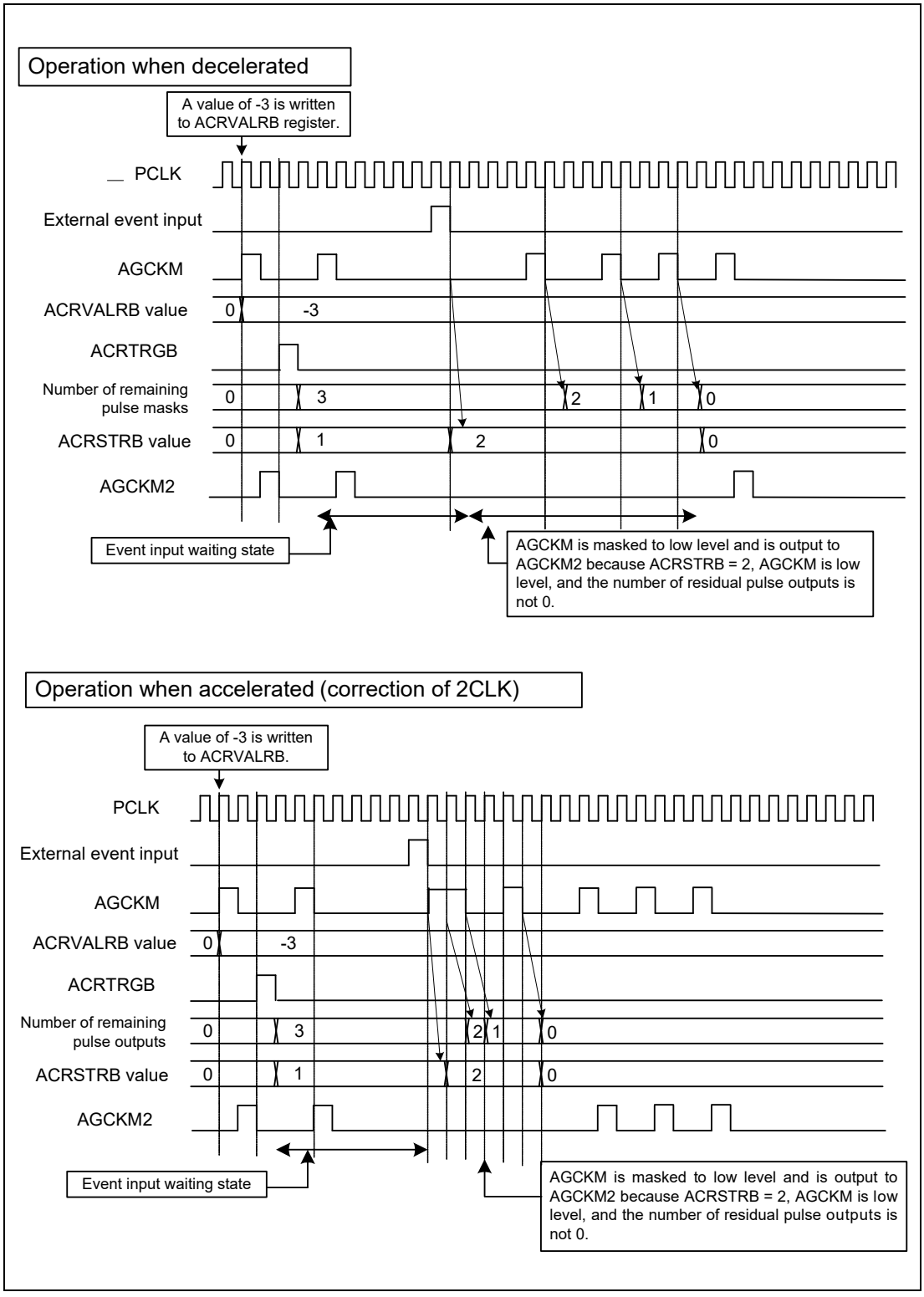


Figure 21.30 AGCKM2 Operation (Negative Logic)

21.6 Timer C

21.6.1 Operation

The timer C block is a general-purpose timer consisting of eight subblocks having the same function.

Each timer C subblock can achieve the following functions.

- Input capture and output compare matches
- Choice of rising edge, falling edge, or both edge sensing as the edge of input capture trigger signal
- Capturing at the timing of event output 1, 2A, or 2B from timer A when used as input capture
- Output of a waveform on compare match. Choice of a logical one, a logical zero, a toggled output, or a one-shot pulse output by setting a register
- Output of an interrupt request and a DMA request on capture or compare match
- Output of an interrupt request on timer counter overflow
- Clearing timer counter Cx (TCNTCx) on compare match (two types)
  - The counter can be cleared on compare match of GRCx0 in PWM mode (not supported by GRCx1 to GRCx3)
  - When the counter upper-limit setting function is used in any mode other than PWM mode, the counter can be cleared on compare match of CUCRCx.
- Output of forced compare match by setting the forced compare match bit
- Noise canceling function (noise canceler count clock or clock bus 5 selectable)

Timer C subblock consists of a 32-bit timer counter Cx (TCNTCx) , four 32-bit timer general registers Cxy (GRCxy), four 32-bit output compare registers Cxy (OCRCxy), a 32-bit counter upper-limit setting compare register Cx (CUCRCx), and controller. The timer general registers Cxy (GRCxy) can be used for input capture/compare match operations and input capture trigger input and output compare output signals (TIOCxy) are available.

The initial output value on the TIOCxy pin is 0 for output compare operation. During operation, the previous state is reflected.

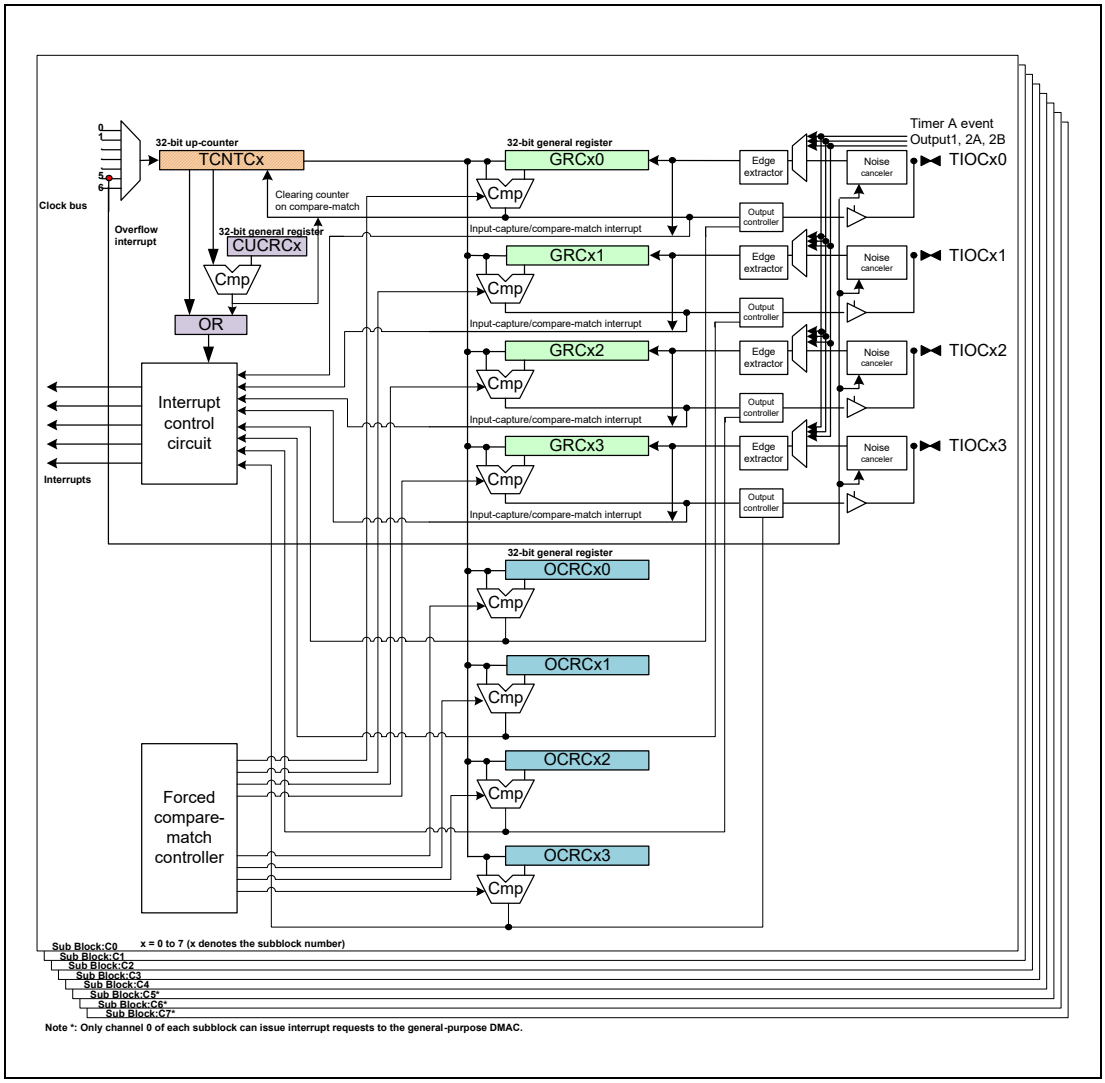


Figure 21.31 Block Diagram of Timer C

21.6.2 Timer C Registers

21.6.2.1 TSTRC — Timer Start Register C

Access: 8-bit accessible

Address: FFE6 0600_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	STRC7	STRC6	STRC5	STRC4	STRC3	STRC2	STRC1	STRC0
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.68 TSTRC Register Contents

Bit Position	Bit Name	Function
7	STRC7	Counter C7 Start 0: TCNTC7 is disabled 1: TCNTC7 is enabled
6	STRC6	Counter C6 Start 0: TCNTC6 is disabled 1: TCNTC6 is enabled
5	STRC5	Counter C5 Start 0: TCNTC5 is disabled 1: TCNTC5 is enabled
4	STRC4	Counter C4 Start 0: TCNTC4 is disabled 1: TCNTC4 is enabled
3	STRC3	Counter C3 Start 0: TCNTC3 is disabled 1: TCNTC3 is enabled
2	STRC2	Counter C2 Start 0: TCNTC2 is disabled 1: TCNTC2 is enabled
1	STRC1	Counter C1 Start 0: TCNTC1 is disabled 1: TCNTC1 is enabled
0	STRC0	Counter C0 Start 0: TCNTC0 is disabled 1: TCNTC0 is enabled

Note 1. Start bits are provided for subblocks C0 to C7 respectively.

TSTRC is an 8-bit readable/writable register that enables and disables timer counter Cx (TCNTCx) in eight subblocks Cx (x = 0 to 7). When the both the STRC bits in this register and the TCE bit in ATU-IV master enable register (ATUENR) are set to 1, counting is enabled.

TSTRC is initialized to 00_H by a reset.

(1) STRCx — Counter Cx Start

These bits enable and disable timer counter Cx (TCNTCx) in a subblock.

When bit STRCx is cleared to 0, TCNTCx is stopped. While TCNTCx is stopped, the previous counter value is retained and TCNTCx is resumed from the value when this bit is set to 1 again.

Even if the counter Cx start bit is set to 1, counting does not start unless the timer Cx master enable bit in the ATU-IV control register is set to 1.

CAUTION

The prescalers run regardless of this bit and are not synchronized with the timing at which this bit is set. Therefore, the time from when this bit is set to 1 until TCNTCx is incremented for the first time is less than the cycle time of the clock of TCNTCx.

21.6.2.2 NCCRCx — Noise Canceler Control Register Cx (x = 0 to 7)

Access: 8-bit accessible

Address: FFE6 0604_H (NCCRC0)
FFE6 0605_H (NCCRC1)
FFE6 0606_H (NCCRC2)
FFE6 0607_H (NCCRC3)
FFE6 0608_H (NCCRC4)
FFE6 0609_H (NCCRC5)
FFE6 $060A_H$ (NCCRC6)
FFE6 $060B_H$ (NCCRC7)

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	NCKCx3	NCKCx2	NCKCx1	NCKCx0	NCECx3	NCECx2	NCECx1	NCECx0
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.69 NCCRCx Register Contents

Bit Position	Bit Name	Function
7 to 4	NCKCxy	Noise Canceler Clock Select These bits select the count source clock for noise canceler counter Cxy (NCNTCxy).
3 to 0	NCECxy	Noise Canceler Enable Cxy These bits set the TIOCxy input noise canceling function.

Note: (x = 0, 1, 2, 3, 4, 5, 6, 7) (y = 0, 1, 2, 3)

Cx (NCCRCx) is an 8-bit readable/writable register. The noise cancellation is performed on the input capture trigger signal input from pin TIOCxy in subblock Cx by setting this register. Three modes are available in noise cancellation and can be switched by the NCMC bit in the noise cancel mode register (NCMR). Noise canceler count clock or clock bus 5 can be selected as the count source by the noise canceler clock select bits (NCKCxy) in this register.

NCCRCx is initialized to 00_H by a reset.

(1) NCKCxy — Noise-Canceler Clock Select

These bits select the count source clock of noise canceler counter Cxy (NCNTCxy).

NCKCxy	Description
0	Noise canceler count clock (Pϕ or Pϕ/128) is selected as the count source clock of NCNTCxy (Initial value).
1	Clock bus 5 is selected as the count source clock of NCNTCxy.

Note: (x = 0, 1, 2, 3, 4, 5, 6, 7) (y = 0, 1, 2, 3)

Noise canceler count clock or clock bus 5 is selectable as the count source clock. As the noise canceler count clock, a clock of Pϕ or Pϕ/128 can be selected by the NCCSEL bit in the common control unit.

(2) NCECxy — Noise Canceler Enable Cxy

These bits enable or disable the noise canceling function for input/output pins TIOCx0 to TIOCx3 of subblock Cx.

NCECxy	Description	
0	The TIOCxy input noise canceling function is disabled	(Initial value).
1	The TIOCxy input noise canceling function is enabled.	

Note: (x = 0, 1, 2, 3, 4, 5, 6, 7) (y = 0, 1, 2, 3)

When a change in input signal level from TIOCxy is detected with these bits set to 1, processing for premature-transition cancellation mode, minimum time-at-level cancellation mode, or level accumulation cancellation mode starts according to the settings of the noise cancellation mode register (NCMR) and the noise canceler clock select bit (NCKC) in the common control unit.

In premature-transition cancellation mode:

When a level change of the externally input signal is detected, the change is output as the signal whose noise is removed and the corresponding noise canceler counter (NCNTCxy) is started for counting up. Subsequent level changes are masked until the value in the counter reaches the value in the noise canceler register (NCRCxy). The level of the externally input signal is output on this compare match.

When these bits are cleared to 0 while the counter (NCNTCxy) is being incremented, counting continues until the values in the counter and the noise canceler register match. The subsequent level changes are also masked over this period.

In minimum time-at-level cancellation mode:

When a level change of the externally input signal is detected, the corresponding noise canceler counter (NCNTCxy) is started for counting up. If subsequent level changes are not detected until the value in the counter reaches the value in the noise canceler register (NCRCxy), the previously accepted level change is output as the signal whose noise is removed on compare match of the counter and noise canceler register.

When the subsequent level change is detected before the values in the counter and noise canceler register match, all the changes are treated as noise. Therefore the signal whose noise is removed is not changed.

When these bits are cleared to 0 while the counter (NCNTCxy) is being incremented, counting continues until the values in the counter and the noise canceler register match or a level change on the externally input signal is detected.

In level accumulation cancellation mode:

The corresponding noise canceler counter Cxy (NCNTCxy) is started for counting up or down according to the input signal level (high level: up-count, low level: down-count). Up-count continues until the noise canceler counter value matches the noise canceler register Cxy (NCRCxy) value. Down-count continues until the noise canceler counter value reaches 0000_H. When a compare match (NCRCxy) occurs during up-counting, the noise canceler output is updated to 1. When a compare match (0000_H) occurs during down-counting, the noise canceler output is updated to 0.

In minimum time-at-level cancellation mode and premature-transition cancellation mode, clock Pϕ is always used to detect input signal level changes regardless of the selected noise canceler clock. In level

accumulation cancellation mode, the clock selected by the noise canceler clock select Cxy bits (NCKCxy) is used for sampling the input level.

For details on operations in each mode, see **Figure 21.1**, **Figure 21.2** and **Figure 21.3**.

21.6.2.3 TCRCx — Timer Control Registers C (x = 0 to 7)

Access: 8-bit/16-bit accessible

Address: FFE6 0624_H (TCRC0)
FFE6 $06A4_H$ (TCRC1)
FFE6 0724_H (TCRC2)
FFE6 $07A4_H$ (TCRC3)
FFE6 0824_H (TCRC4)
FFE6 $08A4_H$ (TCRC5)
FFE6 0924_H (TCRC6)
FFE6 $09A4_H$ (TCRC7)

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	FOCMCx3	FOCMCx2	FOCMCx1	FOCMCx0	—	—	—	CLRCx	FCMCx3	FCMCx2	FCMCx1	FCMCx0	PWMx0	CKSELCx[2:0]		
Value after reset	0*1	0*1	0*1	0*1	0	0	0	0	0*1	0*1	0*1	0*1	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note 1. Writing 0 is disabled. If 1 is written to these bits, data is not retained. These bits are always read as 0.

Table 21.70 TCRCx Register Contents

Bit Position	Bit Name	Function
15 to 12	FOCMCxy	Forced Output Compare Match Cxy Forced compare match is performed by output compare register Cxy (OCRCxy).
11 to 9	—	Not used. These bits are read as 0. When writing, always write 0.
8	CLRCx	TCNTCx Clear Cx Enables or disables clearing of timer counters.
7 to 4	FCMCxy	Forced Compare Match Cxy Forced compare match is performed by timer general register Cxy (GRCxy).
3	PWMx0	PWM Mode Whether to operate subblock Cx in PWM mode is selected.
2 to 0	CKSELCx[2:0]	Clock Select The count clock is selected from clock bus 0 to 6.

Note: x = 0, 1, 2, 3, 4, 5, 6, 7: Corresponding to subblocks C0 to C7
y = 0, 1, 2, 3: Corresponding to general registers Cx0, Cx1, Cx2, and Cx3

The timer control registers Cx (TCRCx) are 8-bit/16-bit readable/writable registers that select the counter clock for subblocks C0 to C7, set operating mode, set forced compare matches, and enable/disable clearing of timer counters.

(1) CKSELCx[2:0] — TCNTCx Clock Select

These bits select the counter clock of subblock Cx.

Timer counters Cx (TCNTCx) in subblock Cx are driven by the clock selected in these bits.

CKSELCx			Description	
0	0	0	Counters are driven by clock-bus line 0	(initial value)
0	0	1	Counters are driven by clock-bus line 1	
0	1	0	Counters are driven by clock-bus line 2	
0	1	1	Counters are driven by clock-bus line 3	
1	0	0	Counters are driven by clock-bus line 4	
1	0	1	Counters are driven by clock-bus line 5	
1	1	0	Counters are driven by clock-bus line 6	
1	1	*	Reserved	

Note: x = 0, 1, 2, 3, 4, 5, 6, 7: Corresponding to subblocks C0 to C7

(2) PWMx0 — PWM Mode

Setting this bit to 1 makes subblock Cn operate in PWM mode. In PWM mode, TCNTCn is cleared on compare match between TCNTCn and timer general register Cx0 (GRCx0). The setting of this bit is valid when GRCn0 functions as a compare match register.

When general register GRCxm (m = 1 to 3) functions as a compare match register, aIOCx0 setting value in the TIORCx is output to its output pins (TIOCx1, TIOCx2, TIOCx3) at the timing of clearing of the TCNTCx.

PWM output from pins TIOCx1 to TIOCx3 is enabled by setting a PWM cycle in GRCx0 and setting a duty cycle in GRCx1 to GRCx3.

CAUTION

To make the subblock operate in PWM mode, further setting is needed. Select the compare match (IOCxy[3:2] = 00_B) by the IOCxy bits in timer I/O control register Cx (TIORCx) for GRCx0 and GRCx1 to GRCx3 (for PWM output). Set IOCxy[1:0] to 01_B or 10_B.

The active polarity of PWM can be specified by the IOCxy[1:0] bits.

PWMx0	Description	
0	0: Subblock Cx does not operate in PWM mode	(initial value)
1	1: Subblock Cx operates in PWM mode	

Note: x = 0, 1, 2, 3, 4, 5, 6, 7: Corresponding to subblocks C0 to C7

When TCNTCx matches GRCn0 while this bit is set to 1, the counter is cleared to 0000 0000_H. However, when clearing the counter on compare match and incrementation occur simultaneously, TCNTCx is set to 0000 0001_H. This occurs when TCNTCx is driven by the clock whose frequency is equal to the Pϕ clock.

In PWM mode, do not set from GRCx0 to GRCx3 to 0000 0000_H. If GRCx0 is set to 0000 0000_H, note that compare match occurs at illegal cycles.

(3) FCMxy — Forced Compare Match Cxy

Setting these bits to 1 generates forced compare match when a general register (Cxy) is used for compare match.

FCMCxy	Description
0	No forced compare match is disabled on timer general register Cxy (GRCxy) (Initial value)
1	Forced compare match generated on general register Cxy (GRCxy)

Note: x = 0, 1, 2, 3, 4, 5, 6, 7: Corresponding to subblocks C0 to C7,
y = 0, 1, 2, 3: Corresponding to general registers Cx0, Cx1, Cx2, and Cx3

Setting the FCMCxy bit to 1 sets the IMFCxy bit (compare match flag) in TSRCx (timer status register Cx) to 1 and outputs a compare match interrupt to the CPU. Even when the IMFCxy bit is set to 1, a compare match interrupt is output. Furthermore, TIOCxy output data also changes as in the case of compare match. This operation is performed only once when the FCMCxy bit is set to 1. Writing 0 to these bits is disabled. If 1 is written to these bits, data is not retained. These bits are always read as 0.

When both PWMx0 bit and FCMCx0 bit are set to 1, the counter is cleared and the set IOCx0 value is output to TIOCxy due to forced compare match. Then, TCNTCx starts counting from 0000 0000_H. (Counter clear due to compare match is enabled only for channel 0 in each subblock.)

(4) CLRCx — TCNTCx Clear Cx

Setting the TCNTCx clear bit Cx to 1 clears the TCNTCx value at the first clock after a compare match between timer counter Cx (TCNTCx) and counter upper-limit setting compare register Cx (CUCRCx).

CLRCx	Description
0	CUCRCx compare match is not used to clear TCNTCx. (Initial value)
1	CUCRCx compare match is used to clear TCNTCx

Note: x = 0, 1, 2, 3, 4, 5, 6, 7: Corresponding to subblocks C0 to C7 and timer counter Cx

The upper-limit setting function (CLRCx = 1) can be set in any mode other than PWM mode (PWMx0 = 0).

When the upper-limit setting function (CLRCx = 1) is set in PWM mode (PWMx0 = 1), timer counter Cx (TCNTCx) is cleared in cycles other than the PWM cycle, which sets the PWM output undefined.

(5) FOCMCxy — Forced Output Compare Match Cxy

When the output compare register Cxy (OCRCxy) is used as a compare match register, a compare match is forcibly generated by setting FOCMCxy to 1.

FOCMCxy	Description
0	Forced compare match is disabled for the output compare register Cxy (OCRCxy). (Initial value)
1	Forced compare match is generated in the output compare register Cxy (OCRCxy).

Note: x = 0, 1, 2, 3, 4, 5, 6, 7: Corresponding to subblocks C0 to C7
y = 0, 1, 2, 3: Corresponding to output compare registers Cx0 to Cx3

Setting the FOCMCxy bit to 1 sets the OCMFCxy bit (compare match flag) in TSRCx (timer status register Cx) to 1 and outputs a compare match interrupt to the CPU. Even when the OCMFCxy bit is 1, a compare match interrupt is generated and the output from TIOCxy changes as when a compare match occurs. This operation is performed only once when 1 is written to this bit. Writing 0 is disabled. If 1 is written to this bit, data is not retained. This bit is always read as 0.

21.6.2.4 TSRCx — Timer Status Registers Cx (x = 0 to 7)

Access: 8-bit/16-bit accessible
Writing 0 is disabled. If 1 is written to these bits, data is not retained. These bits are always read as 0.

Address: FFE6 0626_H (TSRC0)
FFE6 $06A6_H$ (TSRC1)
FFE6 0726_H (TSRC2)
FFE6 $07A6_H$ (TSRC3)
FFE6 0826_H (TSRC4)
FFE6 $08A6_H$ (TSRC5)
FFE6 0926_H (TSRC6)
FFE6 $09A6_H$ (TSRC7)

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	OCMFCx3	OCMFCx2	OCMFCx1	OCMFCx0	—	—	—	OVFCx	IMFCx3	IMFCx2	IMFCx1	IMFCx0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 21.71 TSRCx Register Contents

Bit Position	Bit Name	Function
15 to 12	—	Not used. These bits are read as 0. When writing, always write 0.
11 to 8	OCMFCxy	Output Compare Match Flag Cxy 1: A compare match has occurred by the OCRCxy register. 0: No compare match has occurred by the OCRCxy register. (Initial value)
7 to 3	—	These bits are not used. Fix these bits to 0.
4	OVFCx	Overflow Flag Cx 1: An overflow has occurred. 0: No overflow has occurred.
3 to 0	IMFCxy	Input Capture/Compare Match Flag Cxy 1: Input capture and compare match by the GRCxy register have occurred. 0: No input capture or compare match by the GRCxy register has occurred.

Note: x = 0, 1, 2, 3, 4, 5, 6, 7: Corresponding to subblocks C0 to C7
y = 0, 1, 2, 3: Corresponding to channels 0 to 3

Timer Status Registers Cx (TSRCx) are 8-bit/16-bit read-only registers that indicate occurrence of overflow of timer counter Cx (TCNTCx) in subblocks Cx (x = 0 to 7) and input capture and compare match of timer general registers Cxy (GRCxy, y = 0 to 3), and output compare match of output compare registers Cxy (OCRCxy, y = 0 to 3).

These status flags indicate occurrence of an interrupt request. These flags can be cleared by setting corresponding bits in the timer status clear register Cx (TSCRCx). If an interrupt source occurs with one of these flags set to 1, an interrupt request is generated again. Even when clearing a flag by using the corresponding timer status clear register Cx conflicts with setting the flag due to occurrence of an interrupt source, an interrupt request is generated.

TSRCx is initialized to 0000_H by a reset.

(1) OCMFCxy — Output Compare Match Flag Cxy

This status flag indicates occurrence of an output compare match of the output compare register Cxy (OCRCxy). This flag cannot be set to 1 or 0 by the software. Setting and clearing conditions are described below.

OCMFCxy	Description
0	[Clearing condition: compare match mode/input capture mode/one-shot pulse mode] Writing 1 to OCMFCCxy in timer status clear register Cx (TSCRCx).
1	[Setting conditions: compare match mode/input capture mode] • When the TCNTCx value is equal to the OCRCxy value • Writing 1 to FOCMCxy (forced output compare match bit) in TCRCx [Setting conditions: one-shot pulse mode] • When the TCNTCx value is equal to the OCRCxy value while one-shot pulse is output • Writing 1 to FOCMCxy (forced output compare match bit) in TCRCx while one-shot pulse is output

Note: x = 0, 1, 2, 3, 4, 5, 6, 7: Corresponding to subblocks C0 to C7
y = 0, 1, 2, 3: Corresponding to channels 0 to 3

The next output compare processing is enabled even when the output compare match flag (OCMFCxy) is 1 (the flag is not cleared). At this time, 1 is overwritten to OCMFCxy.

Even if the output compare match status is cleared while the TCNTCx and OCMFCxy values are equal after an output compare match is detected, no status is set newly.

OCMFCxy and IMFCxy shares the same interrupt. Output of interrupt for one or both bits is enabled by setting the timer interrupt enable register Cx (TIERCx).

(2) OVFCx — Overflow Flag Cx

Indicates whether or not TCNTCx has overflowed. This flag cannot be set to 1 or 0 by software.

OVFCx	Description
0	[Clearing condition] Writing 1 to OVFCCx in timer status clear register Cx (TSCRCx) (Initial value)
1	[Setting condition] • When the TCNTCx value overflows (FFFF FFFF$_H$ to 0000 0000$_H$) • When the counter is cleared with the upper-limit setting function enabled

Note: x = 0, 1, 2, 3, 4, 5, 6, 7: Corresponding to subblocks C0 to C7

The overflow flag is set to 1 when the up-count clock is input with TCNTCx set to FFFF FFFF$_H$ or at the next PCLK clock after the counter value matches the upper-limit setting register value with the upper-limit setting function enabled. When writing 0000 0000$_H$ to TCNTCx or

TCNTCx is started from 0000 0000$_H$, this bit is not set to 1.

When writing to TCNTCx at the same time as incrementation while it is FFFF FFFF$_H$, this bit is set to 1. However, TCNTCx is started from the written value.

(3) IMFCxy — Input Capture/Compare Match Flag Cxy

These bits indicate whether or not input capture and compare match between timer general register Cxy (GRCxy) and TCNTCx has occurred. This flag cannot be set to 1 or 0 by software. Setting and clearing conditions are shown below.

IMFCxy	Description
0	[Clearing conditions: compare match mode/input capture mode/one-shot pulse mode] Writing 1 to IMFCCxy in timer status clear register Cx (TSCRCx)
1	[Setting condition: input capture mode] • When GRCxy functions as an input capture register and the value in TCNTCx is transferred to GRCxy on an assertion of the input capture signal [Setting conditions: output compare mode] • When GRCxy functions as a compare match register and the values in TCNTCx and GRCxy match • When the forced compare match bit (FCMCxy) in TCRCx is set to 1 [Setting conditions: one-shot pulse output mode] • When the TCNTCx value is equal to the GRCxy after (or at the same time when) a compare match of OCRCxy is generated while one-shot pulse is output • When the forced compare match bit (FCMCxy) in TCRCx is set to 1 after (or at the same time when) a compare match of OCRCxy is generated while one-shot pulse is output

Note: x = 0, 1, 2, 3, 4, 5, 6, 7: Corresponding to subblocks C0 to C7
y = 0, 1, 2, 3: Corresponding to channels 0 to 3

Even if these bits are set to 1 meaning that the flag has not been cleared, another input capture or output compare signal can be input. A value of 1 is written to these bits.

Even if the compare match flag is cleared to 0 while TCNTCx = IMFCxy after the match in comparison is detected, these bits are not set to 1.

OCMFCxy and IMFCxy shares the same interrupt. Output of interrupt for one or both bits is enabled by setting the timer interrupt enable register Cx (TIERCx).

21.6.2.5 TSCRCx — Timer Status Clear Register Cx (x = 0 to 7)

Access: 8-bit/16-bit accessible

Address: FFE6 0622_H (TSCRC0)
FFE6 $06A2_H$ (TSCRC1)
FFE6 0722_H (TSCRC2)
FFE6 $07A2_H$ (TSCRC3)
FFE6 0822_H (TSCRC4)
FFE6 $08A2_H$ (TSCRC5)
FFE6 0922_H (TSCRC6)
FFE6 $09A2_H$ (TSCRC7)

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	OCMFC Cx3	OCMFC Cx2	OCMFC Cx1	OCMFC Cx0	—	—	—	OVFC Cx	IMFC Cx3	IMFC Cx2	IMFC Cx1	IMFC Cx0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Table 21.72 TSCRCx Register Contents

Bit Position	Bit Name	Function
15 to 12	—	These bits are not used. These bits are read as 0.
11	OCMFCCx3	Output Compare Match Flag Clear Cx3 Enable 0: Disabled (Initial value) 1: OCMFCx3 in timer status register Cx (TSRCx) is cleared to 0.
10	OCMFCCx2	Output Compare Match Flag Clear Cx2 Enable 0: Disabled (Initial value) 1: OCMFCx2 in timer status register Cx (TSRCx) is cleared to 0.
9	OCMFCCx1	Output Compare Match Flag Clear Cx1 Enable 0: Disabled (Initial value) 1: OCMFCx1 in timer status register Cx (TSRCx) is cleared to 0.
8	OCMFCCx0	Output Compare Match Flag Clear Cx0 Enable 0: Disabled (Initial value) 1: OCMFCx0 in timer status register Cx (TSRCx) is cleared to 0.
7 to 5	—	These bits are not used. Fix these bits to 0.
4	OVFCCx	Overflow Flag Clear C Enable 0: Disabled (Initial value) 1: OVFCx in timer status register Cx (TSRCx) is cleared to 0.
3	IMFCCx3	Input Capture Flag Clear C3 Enable 0: Disabled (Initial value) 1: IMFCx3 in timer status register Cx (TSRCx) is cleared to 0.
2	IMFCCx2	Input Capture Flag Clear C2 Enable 0: Disabled (Initial value) 1: IMFCx2 in timer status register C (TSRCx) is cleared to 0.
1	IMFCCx1	Input Capture Flag Clear C1 Enable 0: Disabled (Initial value) 1: IMFCx1 in timer status register Cx (TSRCx) is cleared to 0.
0	IMFCCx0	Input Capture Flag Clear C0 Enable 0: Disabled (Initial value) 1: IMFCx0 in timer status register Cx (TSRCx) is cleared to 0.

Note: x = 0, 1, 2, 3, 4, 5, 6, 7: Corresponding to subblocks C0 to C7

TSCRCx is an 8-bit/16-bit readable/writable register to set clearing of flags at occurrence of an overflow or input capture or compare match.

This register is always read as 0.

TSCRCx is initialized to 0000_H by a reset.

(1) OCMFCCx3 — Output Compare Match Flag Clear Cx3 Enable

Writing 1 to this register while the compare match flag (OCMFCx3) in the timer status register Cx (TSRCx) is set to 1 clears OCMFCx3 to 0. This bit is always read as 0.

(2) OCMFCCx2 — Output Compare Match Flag Clear Cx2 Enable

Writing 1 to this register while the compare match flag (OCMFCx2) in the timer status register Cx (TSRCx) is set to 1 clears OCMFCx2 to 0. This bit is always read as 0.

(3) OCMFCCx1 — Output Compare Match Flag Clear Cx1 Enable

Writing 1 to this register while the compare match flag (OCMFCx1) in the timer status register Cx (TSRCx) is set to 1 clears OCMFCx1 to 0. This bit is always read as 0.

(4) OCMFCCx0 — Output Compare Match Flag Clear Cx0 Enable

Writing 1 to this register while the compare match flag (OCMFCx0) in the timer status register Cx (TSRCx) is set to 1 clears OCMFCx0 to 0. This bit is always read as 0.

(5) OVFCCx — Overflow Flag Clear Cx Enable

Writing 1 to this bit while overflow flag Cx (OVFCx) in the timer status register Cx (TSRCx) is set to 1 clears OVFCx to 0. This bit is always read as 0.

(6) IMFCCx3 — Input Capture Flag Clear Cx3 Enable

Writing 1 to this bit while input capture flag Cx3 (IMFCx3) in the timer status register Cx (TSRCx) is set to 1 clears IMFCx3 to 0. This bit is always read as 0.

(7) IMFCCx2 — Input Capture Flag Clear Cx2 Enable

Writing 1 to this bit while input capture/compare match flag Cx2 (IMFCx2) in the timer status register Cx (TSRCx) is set to 1 clears IMFCx2 to 0. This bit is always read as 0.

(8) IMFCCx1 — Input Capture Flag Clear Cx1 Enable

Writing 1 to this bit while input capture/compare match flag Cx1 (IMFCx1) in the timer status register Cx (TSRCx) is set to 1 clears IMFCx1 to 0. This bit is always read as 0.

(9) IMFCCx0 — Input Capture Flag Clear Cx0 Enable

Writing 1 to this bit while input capture/compare match flag Cx0 (IMFCx0) in the timer status register Cx (TSRCx) is set to 1 clears IMFCx0 to 0. This bit is always read as 0.

21.6.2.6 TIORCx — Timer I/O Control Registers Cx (x = 0 to 7)

Access: 8-bit and 16-bit accessible

Address: FFE6 0628_H (TIORC0)
FFE6 $06A8_H$ (TIORC1)
FFE6 0728_H (TIORC2)
FFE6 $07A8_H$ (TIORC3)
FFE6 0828_H (TIORC4)
FFE6 $08A8_H$ (TIORC5)
FFE6 0928_H (TIORC6)
FFE6 $09A8_H$ (TIORC7)

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IOCx3[3:0]				IOCx2[3:0]				IOCx1[3:0]				IOCx0[3:0]			
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.73 TIORCx Register Contents

Bit Position	Bit Name	Function
15 to 0	IOCxy[3:0]	Settings of timer general register Cxy (GRCxy) functions, input capture and compare match, edge to be extracted at input capture, and output value at compare match

Note: x = 0, 1, 2, 3, 4, 5, 6, 7: Corresponding to subblocks C0 to C7
y = 0, 1, 2, 3: Corresponding to channels 0 to 3

(1) IOCxy[3:0] — I/O Control bit

IOCxy					
[3]	[2]	[1]	[0]		Function
0	0	0	0	Compare match mode GRCxy is used as output compare register. OCRCxy is used as output compare register.	Compare match by GRCxy disabled (Initial value) Compare match by OCRCxy enabled
0	0	0	1		A value of 0 is output at compare match by GRCxy Compare match by OCRCxy enabled
0	0	1	0		A value of 1 is output at compare match by GRCxy Compare match by OCRCxy enabled
0	0	1	1		Toggle output at compare match by GRCxy Compare match by OCRCxy enabled
0	1	0	0	Input capture mode GRCxy is used as input capture register. OCRCxy is used as output compare register.	Input capture disabled Compare match by OCRCxy enabled
0	1	0	1		Capturing at the rising edge of TIOCxy Compare match by OCRCxy enabled
0	1	1	0		Capturing at the falling edge of TIOCxy Compare match by OCRCxy enabled
0	1	1	1		Capturing at the rising edge and falling edge of TIOCxy Compare match by OCRCxy enabled
1	0	0	0	One-shot pulse mode GRCxy and OCRCxy are used as output compare register for one-shot.*2	Compare match of OCRCxy and GRCxy disabled A value of 0 is output from TIOCxy output (active high)
1	0	0	1		Compare match of OCRCxy and GRCxy disabled A value of 1 is output from TIOCxy output (active low)
1	0	1	0		A value of 0 is output from TIOCxy output until occurrence of a compare match of OCRCxy, a value of 1 is output from TIOCxy output from occurrence of a compare match of OCRCxy until occurrence of a compare match of GRCxy, and then a value of 0 is output from TIOCxy output at occurrence of a compare match of GRCxy (active high). *1
1	0	1	1		A value of 1 is output from TIOCxy output until occurrence of a compare match of OCRCxy, a value of 0 is output from TIOCxy output from occurrence of a compare match of OCRCxy until occurrence of a compare match of GRCxy, and then a value of 1 is output from TIOCxy output at occurrence of a compare match of GRCxy (active low) *1
1	1	0	0	Input capture mode GRCxy is used as input capture register. OCRCxy is used as output compare register.	Input capture disabled Compare match by OCRCxy enabled
1	1	0	1		Capturing by timer A event output 1 Compare match by OCRCxy enabled
1	1	1	0		Capturing by timer A event output 2A Compare match by OCRCxy enabled
1	1	1	1		Capturing by timer A event output 2B Compare match by OCRCxy enabled

Note: x = 0, 1, 2, 3, 4, 5, 6, 7) (y = 0, 1, 2, 3)

Note 1. If a compare match by OCRCxy and a compare match by GRCxy occur at the same time, the compare match by GRCxy takes precedence.

Note 2. Before selecting one-shot pulse mode (IOCxy[3:2] = 10_B), set the PWMX0 bit to 0. It is prohibited to set PWMX0 bit to 1.

The IOCxy[3:2] bits are used to set operating mode (compare match mode, input capture mode, or one-shot pulse mode). Change operating mode (IOCxy[3:2]) while the counter is not working.

Disable compare match and input capture (by setting IOCxy[1:0] to 00_B) before switching output mode for compare match mode (IOCxy[3:2] = 00_B) or switching edge detection/trigger for input capture mode (IOCxy[3:2] = 01_B or 11_B).

Change operating mode to one-shot pulse mode (IOCxy[3:2] = 10_B) and change active polarity (IOCxy[1:0]) in one-shot pulse mode (IOCxy[3:2] = 10_B) while the counter is not working.

When one-shot pulse mode (IOCxy[3:2] = 10_B) is used, initialize the external pins (TIOCxy) by forced compare match Cxy (FCMCxy) after operating mode is set, and then start counting.

21.6.2.7 TCNTCx — Timer Counters Cx (x = 0 to 7)

Access: 32-bit accessible but 8-bit/16-bit inaccessible

Address: FFE6 0638_H (TCNTC0)
FFE6 $06B8_H$ (TCNTC1)
FFE6 0738_H (TCNTC2)
FFE6 $07B8_H$ (TCNTC3)
FFE6 0838_H (TCNTC4)
FFE6 $08B8_H$ (TCNTC5)
FFE6 0938_H (TCNTC6)
FFE6 $09B8_H$ (TCNTC7)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CNTC[31:16]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CNTC[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Timer counter Cx (TCNTCx) is 32-bit readable/writable registers driven by the input clock. These counters can be read from and written to while they are being run.

Timer counter Cx (TCNTCx) is started for counting by setting the bit in timer start register C (TSTRC) to 1. The clock signal is selected by the clock select bit (CKSEL) in timer control register Cx (TCRCx). A timer overflow causes an overflow interrupt request to be output to the CPU. TCNTCx is initialized to $0000\ 0000_H$ by a reset.

21.6.2.8 GRCxy — General Registers Cxy

(x = 0, 1, 2, 3, 4, 5, 6, 7: Corresponding to subblocks C0 to C7, y = 0, 1, 2, 3: Corresponding to channels 0 to 3)

Access: 32-bit accessible but 8-bit/16-bit inaccessible

Address: FFE6 0640_H + (80_H * x) + (4_H * y)

Value after reset: FFFF $FFFF_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GRCxy[31:16]															
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GRCxy[15:0]															
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

These timer general registers Cxy (GRCxy) are 32-bit readable/writable registers that function as the input capture register or output compare register. These functions are switched by setting timer I/O control register Cx (TIORCx).

While a timer general register Cx is used as an input capture register, it detects an external input capture signal, stores the TCNTCx value, and outputs an input capture interrupt request to the CPU and a start request to the DMA. At this time, the IMFCxy bit in timer status register Cx (TSRCx) is set to 1. The edge to be detected is selected by TIORCx.

Input capture is performed even if the counter is stopped (the TCE bit in ATUENR is cleared to 0 or the STRCx bit in TSTRC is cleared to 0). The value in the counter stopped is loaded to GRCxy.

If these registers function as the output compare register for compare match, the values in GRCxy and the timer counter Cx (TCNTCx) are constantly compared. When these values match, TIOCxy output is changed according to the method (0 output, 1 output, or toggle output) specified in TIORCx, a compare match interrupt request is output to the CPU, and a start request is output to the DMA. At this time, the IMFCxy bit in the timer status register Cx (TSRCx) is set to 1. Initially, logical zero is output on the TIOCxy pin (immediately after a reset or output compare modes are switched). During operation, the previous value is output.

If these registers function as the output compare register for one-shot pulse output, one-shot pulse output starts by writing to GRCxy. Set OCRCxy before writing to GRCxy.

When one-shot pulse output starts, the values in GRCxy and timer counter Cx (TCNTCx) are compared after the match in comparison with OCRCxy is detected (or when GRCxy = OCRCxy).

Compare match of GRCxy inactivates one-shot pulse (compare match with OCRCxy activates one-shot pulse) and the IMFCxy bit in the timer status register Cx (TSRCx) is set to 1.

Once the one-shot pulse is output, another output is not performed until new value is set to GRCxy. Also, the IMFCxy bit in the timer status register Cx (TSRCx) is not changed.

When these registers are used as output compare registers for compare match or one-shot pulse output, if the counter upper-limit setting function (see **Section 21.6.3.5, Counter Upper-Limit Setting Function**) is in use with clock-bus line 5 or 6, or with a division ratio of 1/1 selected for the prescaler, the timer counter counts between 1 and the value of CUCRCx and the counter value may not become $0000\ 0000_H$. Therefore, do not set these registers to $0000\ 0000_H$ in such cases.

GRCxy is initialized to $FFFF\ FFFF_H$ by a reset.

21.6.2.9 NCNTCxy — Noise Canceler Counters Cxy

(x = 0, 1, 2, 3, 4, 5, 6, 7: Corresponding to subblocks C0 to C7, y = 0, 1, 2, 3: Corresponding to channels 0 to 3)

Access: 16-bit accessible but 8-bit inaccessible

Address: FFE6 0670_H + (80_H * x) (NCNTCx0)
FFE6 0674_H + (80_H * x) (NCNTCx1)
FFE6 0678_H + (80_H * x) (NCNTCx2)
FFE6 $067C_H$ + (80_H * x) (NCNTCx3)

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	NCNTCxy															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.74 NCNTCxy Register Contents

Bit Position	Bit Name	Function
15 to 0	NCNTCxy	Noise Cancel Count Cxy These bits store a 16-bit count value

Noise Canceler counters Cxy (NCNTCx0 to x3) are 8-bit readable/writable registers.

NCNTCx0 to x3 are started by the external input pins (TIOCx0 to TIOCx3) as triggers when the noise canceler is enabled by the noise cancel enable bit (NCECx0 to x3) in timer I/O control register Cx (TIORCx). In level accumulation cancellation mode, these counters increment or decrement according to the external input level. Up-count or down-count after operation start is performed in synchronization with the noise canceler count clock or clock bus 5 supplied from the prescaler.

NCNTCxy continues counting regardless of the settings in the timer C enable bit (TCE) in ATU-IV master enable register (ATUENR) and in TSTRC (regardless of the TCNTCx).

NCNTCxy operates in one of three modes (premature-transition cancellation mode, minimum time-at-level cancellation mode, or level accumulation cancellation mode) according to the settings of the noise cancellation mode select bit (NCMSEL) and the timer C noise cancellation mode bit (NCMC) in the noise cancellation mode register (NCMR) in the common control unit.

- Premature-transition cancellation mode
  When a level change of the externally input signal (TIOCxy) is detected while bit NCECxy is set to 1 and NCNTCxy is stopped, NCNTCxy is started for counting up. This counter is cleared to 0000_H and stopped on the first edge of the PCLK after the value in NCNTCxy matches the value in noise cancel register Cxy (NCRCxy).
  NCNTCxy is incremented regardless of the TCE bit in ATUENR.
  The first change is output as the signal whose noise is removed and the edge is to be extracted. Subsequent level changes are masked until the value in the counter reaches the value in the noise cancel register (NCRCxy). The level of the externally input signal is output on this compare match.

  When the NCECxy bit is cleared to 0 while the counter (NCRCxy) is being incremented, counting continues until the values in the counter and the noise canceler register match. The subsequent level changes are also masked over this period.

- In minimum time-at-level cancellation mode
  When a level change of the externally input signal (TIOCxy) is detected while the NCECxy bit is set to 1 and NCNTxy is stopped, NCNTxy is started for counting up. This counter is cleared to 0000_H and stopped on the first edge of the PCLK after the value in NCNTxy matches the value in noise cancel registers Cxy (NCRCxy) or after the level of the externally input signal (TIOCxy) is changed.
  NCNTxy is incremented regardless of the TCE bits in ATUENR.
  When a level change of the externally input signal is detected, the corresponding noise canceler counter (NCNTxy) is started for counting up. If subsequent level changes are not detected until the value in the counter reaches the value in the noise canceler register (NCRCxy), the previously accepted level change is output as the signal whose noise is removed on compare match of the counter and noise canceler register. When the subsequent level change is detected before the values in the counter and noise canceler register match, all the changes is treated as noise. Therefore the signal whose noise is removed is not changed.
  When the NCECxy bits are cleared to 0 while the counter (NCNTCxy) is being incremented, counting continues until the values in the counter and the noise canceler register match or a level change on the externally input signal is detected.

- In level accumulation cancellation mode
  When the NCECxy bit is 1, NCNTCxy increments or decrements according to the input signal level. input While the input level is high, NCNTCxy increments. When the counter value matches the noise cancel register Cxy (NCRCxy) value, up-counting stops on the next PCLK. While the input level is low, NCNTCxy decrements. When the counter value reaches 0000_H, down-counting stops on the next PCLK.
  NCNTCxy counts regardless of the TCE bit value in the ATU-IV master enable register (ATUENR).
  When the counter value matches the NCRCxy value during up-counting, the noise canceler output is updated to 1. When the counter value reaches 0000_H during down-counting, the noise canceler output is updated to 0.
  When the NCECxy bit is cleared during counting, the noise canceler counter stops working and the value changes from the noise canceler output to the input signal level at that time. For this reason, note that when clearing the NCECxy bit in level accumulation cancellation mode, edge detection may be made at this changeover.
  NCNTCxy is initialized to 0000_H by a reset.

21.6.2.10 NCRCxy — Noise Canceler Registers Cxy

(x = 0, 1, 2, 3, 4, 5, 6, 7: Corresponding to subblocks C0 to C7, y = 0, 1, 2, 3: Corresponding to channels 0 to 3)

Access: 16-bit accessible but 8-bit inaccessible

Address: FFE6 0672_H + (80_H * x) (NCRCx0)
FFE6 0676_H + (80_H * x) (NCRCx1)
FFE6 $067A_H$ + (80_H * x) (NCRCx2)
FFE6 $067E_H$ + (80_H * x) (NCRCx3)

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	NCRCxy															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.75 NCRCxy Register Contents

Bit Position	Bit Name	Function
15 to 0	NCRCxy	Noise Canceling Time Cxy TIOCxy noise canceling period (16-bit comparison value)

Noise cancel resisters Cxy (NCSCx0 to Cx3) are 16-bit readable/writable registers that are provided in each subblock and set the upper limitations of noise cancel counters Cxy (NCNTCx0 to Cx3).

When PCLK/128 clock is selected as a noise canceler clock, noise with maximum length of 0.21 s (at PCLK = 40 MHz) can be canceled when this register is set to $FFFF_H$.

NCSCx0 to NCSCx3 operate in one of three modes (premature-transition cancellation mode, minimum time-at-level cancellation mode, or level accumulation cancellation mode) according to the settings of the noise cancellation mode select bit (NCMSEL) and the timer C noise cancellation mode bit (NCMC) in the noise cancellation mode register (NCMR) in the common control unit.

(1) Premature-Transition Cancellation Mode

While NCNTCxy is in count operation, the level change of the subsequent input signal is masked. Values in NCNTCxy and NCRCxy are always compared. If a compare match occurs, the value in NCNTCxy is cleared on the next PCLK, the count operation is stopped, and the masking of the input signal is canceled.

(2) In Minimum Time-at-Level Cancellation Mode

While NCNTCxy is in count operation, noise canceler processing waiting state is entered. Values in NCNTCxy and NCRCxy are always compared. If a compare match occurs, the value in NCNTCxy is cleared on the next PCLK, the count operation is stopped, and at the same time the noise canceler outputs the input signal that has passed through the noise canceling processing.

(3) In Level Accumulation Cancellation Mode

While NCNTCxy is counting up, the NCNTCxy value is compared with the NCRCxy value. When a compare match occurs, NCNTCxy stops up-counting on the next PCLK. While NCNTCxy is counting down, the NCNTCxy value is compared with 0000_H.

NCRCx0 to NCRCx3 can be read and written only in 16-bit units.

NCRCx0 to NCRCx3 are initialized to 0000_H by a reset.

21.6.2.11 OCRCxy — Output Compare Registers Cxy

(x = 0, 1, 2, 3, 4, 5, 6, 7: Corresponding to subblocks C0 to C7, y = 0, 1, 2, 3: Corresponding to channels 0 to 3)

Access: 32-bit accessible but 8-bit/16-bit inaccessible

Address: FFE6 0650_H + (80_H * x) + (4_H * y)

Value after reset: FFFF $FFFF_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	OCRCxy[31:16]															
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OCRCxy[15:0]															
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Output compare registers Cxy (OCRCxy) are 32-bit readable/writable registers that function as an output compare register.

These registers are used as a compare register in modes other than one-shot pulse mode.

The values in OCRCxy and timer counter Cx (TCNTCx) are always compared. When these values match, a compare match interrupt request and DMA activation request are output to the CPU. At this time, the OCMFCxy bit in the timer status register Cx (TSRCx) is set to 1.

If these registers function as the output compare register for one-shot pulse output, one-shot pulse output starts by writing to GRCxy. Set OCRCxy before writing to GRCxy.

When one-shot pulse output starts, the values in OCRCxy and timer counter Cx (TCNTCx) are compared.

Compare match with OCRCxy activates one-shot pulse (compare match with GRCxy inactivates one-shot pulse) and the OCMFCxy bit in the timer status register Cx (TSRCx) is set to 1.

Once the one-shot pulse is output, another output is not performed until new value is set to GRCxy. Also, the OCMFCxy bit in the timer status register Cx (TSRCx) is not changed.

When these registers are used as output compare registers for compare match or one-shot pulse output, if the counter upper-limit setting function (see **Section 21.6.3.5, Counter Upper-Limit Setting Function**) is in use with clock-bus line 5 or 6, or with a division ratio of 1/1 selected for the prescaler, the timer counter counts between 1 and the value of CUCRCx and the counter value may not become 0000 0000_H. Therefore, do not set these registers to 0000 0000_H in such cases.

OCRCxy is initialized to FFFF $FFFF_H$ by a reset.

21.6.2.12 TIERCx — Timer Interrupt Enable Registers Cx

(x = 0, 1, 2, 3, 4, 5, 6, 7: Corresponding to subblocks C0 to C7)

Access: 8-bit/16-bit accessible

Address: FFE6 0634_H (TIERC0)
FFE6 $06B4_H$ (TIERC1)
FFE6 0734_H (TIERC2)
FFE6 $07B4_H$ (TIERC3)
FFE6 0834_H (TIERC4)
FFE6 $08B4_H$ (TIERC5)
FFE6 0934_H (TIERC6)
FFE6 $09B4_H$ (TIERC7)

Value after reset: $000F_H$

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	OCRCEx3	OCRCEx2	OCRCEx1	OCRCEx0	—	—	—	—	GRCEx3	GRCEx2	GRCEx1	GRCEx0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 21.76 TIERCx Register Contents

Bit Position	Bit Name	Function
15 to 12	—	Not used. These bits are read as 0.
11 to 8	OCRCEx[3:0]	OCRC Interrupt Enable xy Interrupt request due to comparison by OCRCxy 0: Interrupt request due to comparison by OCRCxy is disabled. 1: Interrupt request due to comparison by OCRCxy is enabled.
7 to 4	—	Not used. These bits are read as 0.
3 to 0	GRCEx[3:0]	GRC Interrupt Enable xy 0: Interrupt request due to comparison and capture by GRCxy is disabled. 1: Interrupt request due to comparison and capture by GRCxy is enabled.

Timer interrupt enable registers Cx (TIRECx) are 16-bit readable/writable registers used to set interrupt requests due to comparison in subblocks C0 to C7.

(1) OCRCExy — Interrupt request Cxy due to comparison

An interrupt is generated on compare match by the output compare register Cxy (OCRCxy).

(2) GRCExy — Interrupt request Cxy due to comparison and capture

An interrupt is generated on compare match or capture operation by the general register GRCxy.

21.6.2.13 CUCRCx — Counter Upper-Limit Setting Compare Registers Cx

(x = 0, 1, 2, 3, 4, 5, 6, 7: Corresponding to subblocks C0 to C7)

Access: 32-bit accessible but 8-bit/16-bit inaccessible

Address: FFE6 063C$_H$ (CUCRC0)
FFE6 06BC$_H$ (CUCRC1)
FFE6 073C$_H$ (CUCRC2)
FFE6 07BC$_H$ (CUCRC3)
FFE6 083C$_H$ (CUCRC4)
FFE6 08BC$_H$ (CUCRC5)
FFE6 093C$_H$ (CUCRC6)
FFE6 09BC$_H$ (CUCRC7)

Value after reset: FFFF FFFF$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CUCRCx[31:16]															
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CUCRCx[15:0]															
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Counter upper-limit value compare registers Cx (CUCRCx) are 32-bit readable/writable registers that has a function to be compared with the timer counter Cx (TCNTCx). Matching in the comparison between this register and the timer counter Cx (TCNTCx) is enabled by setting the TCNTCx clear bit Cx (CLRCx) in the timer control register Cx (TCRCx) to 1.

The timer counter Cx (TCNTCx) is cleared to 0000 0000$_H$ in response to a match in comparison between the timer counter Cx (TCNTCx) and counter upper-limit setting compare register Cx (CUCRCx). If this match in comparison occurs at the same time as counting up by the timer counter Cx (TCNTCx), the timer counter Cx (TCNTCx) is cleared to 0000 0001$_H$.

The OVFCx bit in timer status register Cx (TSRCx) is set to 1 and an overflow interrupt is output on compare match between the timer counter Cx (TCNTCx) and counter upper-limit setting compare register Cx (CUCRCx).

Do not set CUCRCx to 0000 0000$_H$. Note that if CUCRCx is set to 0000 0000$_H$, a compare match occurs at incorrect intervals.

NOTE

The upper-limit setting function (CLRCx = 1) can be set in modes other than PWM mode (PWMx0 = 0).

21.6.2.14 NCMCR1C — Noise Cancellation Mode Channel Register 1C

Access: 8-bit accessible

Address: FFE6 0610_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	NCM1C7	NCM1C6	NCM1C5	NCM1C4	NCM1C3	NCM1C2	NCM1C1	NCM1C0
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.77 NCMCR1C Register Contents

Bit Position	Bit Name	Function
7	NCM1C7	Subblock 7 Noise Cancellation Mode This bit specifies the operating mode of Subblock 7 noise canceler. 0: Premature-transition cancellation mode 1: Minimum time-at-level cancellation mode (When NCMC = 0 and NCM2C7 = 0) 1: Level accumulation cancellation mode (When NCMC = 0 and NCM2C7 = 1)
6	NCM1C6	Subblock 6 Noise Cancellation Mode This bit specifies the operating mode of Subblock 6 noise canceler. 0: Premature-transition cancellation mode 1: Minimum time-at-level cancellation mode (When NCMC = 0 and NCM2C6 = 0) 1: Level accumulation cancellation mode (When NCMC = 0 and NCM2C6 = 1)
5	NCM1C5	Subblock 5 Noise Cancellation Mode This bit specifies the operating mode of subblock 5 noise canceler. 0: Premature-transition cancellation mode 1: Minimum time-at-level cancellation mode (When NCMC = 0 and NCM2C5 = 0) 1: Level accumulation cancellation mode (When NCMC = 0 and NCM2C5 = 1)
4	NCM1C4	Subblock 4 Noise Cancellation Mode This bit specifies the operating mode of subblock 4 noise canceler. 0: Premature-transition cancellation mode 1: Minimum time-at-level cancellation mode (When NCMC = 0 and NCM2C4 = 0) 1: Level accumulation cancellation mode (When NCMC = 0 and NCM2C4 = 1)
3	NCM1C3	Subblock 3 Noise Cancellation Mode This bit specifies the operating mode of subblock 3 noise canceler. 0: Premature-transition cancellation mode 1: Minimum time-at-level cancellation mode (When NCMC = 0 and NCM2C3 = 0) 1: Level accumulation cancellation mode (When NCMC = 0 and NCM2C3 = 1)
2	NCM1C2	Subblock 2 Noise Cancellation Mode This bit specifies the operating mode of subblock 2 noise canceler. 0: Premature-transition cancellation mode 1: Minimum time-at-level cancellation mode (When NCMC = 0 and NCM2C2 = 0) 1: Level accumulation cancellation mode (When NCMC = 0 and NCM2C2 = 1)
1	NCM1C1	Subblock 1 Noise Cancellation Mode This bit specifies the operating mode of subblock 1 noise canceler. 0: Premature-transition cancellation mode 1: Minimum time-at-level cancellation mode (When NCMC = 0 and NCM2C1 = 0) 1: Level accumulation cancellation mode (When NCMC = 0 and NCM2C1 = 1)
0	NCM1C0	Subblock 0 Noise Cancellation Mode This bit specifies the operating mode of subblock 0 noise canceler. 0: Premature-transition cancellation mode 1: Minimum time-at-level cancellation mode (When NCMC = 0 and NCM2C0 = 0) 1: Level accumulation cancellation mode (When NCMC = 0 and NCM2C0 = 1)

The noise cancellation mode channel register 1C is an 8-bit readable/writable register used to select an operating mode of noise canceler in each channel.

Premature-transition cancellation mode detects a change in input signal level, and then ignores changes in input signal level within the specified time period. Changes in input signal level within the specified time period after the first level change are regarded as noise in this mode.

Level accumulation cancellation mode accumulates input signal levels and regards that the input level has reached 0 or 1 when the accumulation result becomes 0 or the specified value.

Each period is specified by the noise cancellation register in each channel, and the noise cancellation counter measures the time.

Figure 21.1, **Figure 21.2**, and **Figure 21.3** show schematics of noise cancellation operation (example of TIA00 input of timer A) in premature-transition cancellation mode, minimum time-at-level cancellation mode, and level accumulation cancellation mode, respectively.

Edge detection is made for signals that have passed through the noise canceler in each channel. **Figure 21.1**, **Figure 21.2**, and **Figure 21.3** show examples of rising-edge detection of signals that have passed through the noise canceler.

(1) NCM1C7 — Subblock 7 Noise Cancellation Mode

This bit specifies the operating mode of subblock 7 noise canceler.

(2) NCM1C6 — Subblock 6 Noise Cancellation Mode

This bit specifies the operating mode of subblock 6 noise canceler.

(3) NCM1C5 — Subblock 5 Noise Cancellation Mode

This bit specifies the operating mode of subblock 5 noise canceler.

(4) NCM1C4 — Subblock 4 Noise Cancellation Mode

This bit specifies the operating mode of subblock 4 noise canceler.

(5) NCM1C3 — Subblock 3 Noise Cancellation Mode

This bit specifies the operating mode of subblock 3 noise canceler.

(6) NCM1C2 — Subblock 2 Noise Cancellation Mode

This bit specifies the operating mode of subblock 2 noise canceler.

(7) NCM1C1 — Subblock 1 Noise Cancellation Mode

This bit specifies the operating mode of subblock 1 noise canceler.

(8) NCM1C0 — Subblock 0 Noise Cancellation Mode

This bit specifies the operating mode of subblock 0 noise canceler.

CAUTIONS

1. **Setting this register is enabled only when the NCMC bit in the noise cancellation mode register (NCMR) of the common controller is 0. Furthermore, in case the noise cancellation mode bit of corresponding channel (NCM1Cx) in this register is set to 1, the mode can be set to premature-transition cancellation mode or minimum time-at-level cancellation mode depending on the status of the corresponding channel bit in the noise cancellation mode channel register 2C (NCMCR2C).**
2. **x = 0 to 7**

Table 21.78 shows the truth table for settings that determine noise cancellation modes.

Table 21.78 Truth Table for Noise Cancellation Modes of Timer C

Channel enable	Filter mode		Channel register			
NCCRCx NCECx	NCMR. NCMC	NCMR. NCMSEL	NCMCR1C. NCM1Cx	NCMCR2C. NCM2Cx	Operating mode	Unit of filtering
0	—	—	—	—	Filter invalid	—
1	0	(—)	0	(—)	Premature-transition	Each subblock/ channel
1	0	(—)	1	0	Minimum time-at-level	Each channel
1	0	(—)	1	1	Level accumulation	Each channel
1	1	(0)	—	—	Minimum time-at-level	Each subblock
1	1	1	—	—	Level accumulation	Each subblock

Note 1. x = 0 to 7

Note 2. An enable bit (NCECx) can be set for each channel.

Setting when each channel is the unit of filtering of timer C (precondition: channel enable = 1)

- Setting each channel

  When the noise cancellation mode bit (NCMC) in the noise cancellation mode register (NCMR) is set to 0 and any of the noise cancellation mode channel setting bits (NCM1Cx) in the noise cancellation mode channel register 1C (NCMCR1C) of timer C is set to 1

- Setting all channels

  When the noise cancellation mode bit (NCMC) in the noise cancellation mode register (NCMR) is 1 in the description of “Setting each channel” above or when all of the noise cancellation mode channel setting bits (NCM1Cx) in the noise cancellation mode channel register 1C (NCMCR1C) are set to 0 in the description of “Setting each channel” above

21.6.2.15 NCMCR2C — Noise Cancellation Mode Channel Register 2C

Access: 8-bit accessible

Address: FFE6 0612_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	NCM2C7	NCM2C6	NCM2C5	NCM2C4	NCM2C3	NCM2C2	NCM2C1	NCM2C0
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.79 NCMCR2C Register Contents

Bit Position	Bit Name	Function
7	NCM2C7	Subblock 7 Noise Cancellation Mode This bit specifies the operating mode of Subblock 7 noise canceler. 0: Minimum time-at-level cancellation mode (When NCMC = 0 and NCM1C7 = 1) 1: Level accumulation cancellation mode (When NCMC = 0 and NCM1C7 = 1)
6	NCM2C6	Subblock 6 Noise Cancellation Mode This bit specifies the operating mode of Subblock 6 noise canceler. 0: Minimum time-at-level cancellation mode (When NCMC = 0 and NCM1C6 = 1) 1: Level accumulation cancellation mode (When NCMC = 0 and NCM1C6 = 1)
5	NCM2C5	Subblock 5 Noise Cancellation Mode This bit specifies the operating mode of subblock 5 noise canceler. 0: Minimum time-at-level cancellation mode (When NCMC = 0 and NCM1C5 = 1) 1: Level accumulation cancellation mode (When NCMC = 0 and NCM1C5 = 1)
4	NCM2C4	Subblock 4 Noise Cancellation Mode This bit specifies the operating mode of subblock 4 noise canceler. 0: Minimum time-at-level cancellation mode (When NCMC = 0 and NCM1C4 = 1) 1: Level accumulation cancellation mode (When NCMC = 0 and NCM1C4 = 1)
3	NCM2C3	Subblock 3 Noise Cancellation Mode This bit specifies the operating mode of subblock 3 noise canceler. 0: Minimum time-at-level cancellation mode (When NCMC = 0 and NCM1C3 = 1) 1: Level accumulation cancellation mode (When NCMC = 0 and NCM1C3 = 1)
2	NCM2C2	Subblock 2 Noise Cancellation Mode This bit specifies the operating mode of subblock 2 noise canceler. 0: Minimum time-at-level cancellation mode (When NCMC = 0 and NCM1C2 = 1) 1: Level accumulation cancellation mode (When NCMC = 0 and NCM1C2 = 1)
1	NCM2C1	Subblock 1 Noise Cancellation Mode This bit specifies the operating mode of subblock 1 noise canceler. 0: Minimum time-at-level cancellation mode (When NCMC = 0 and NCM1C1 = 1) 1: Level accumulation cancellation mode (When NCMC = 0 and NCM1C1 = 1)
0	NCM2C0	Subblock 0 Noise Cancellation Mode This bit specifies the operating mode of subblock 0 noise canceler. 0: Minimum time-at-level cancellation mode (When NCMC = 0 and NCM1C0 = 1) 1: Level accumulation cancellation mode (When NCMC = 0 and NCM1C0 = 1)

The noise cancellation mode channel register 2C is an 8-bit readable/writable register used to select an operating mode of noise canceler in each channel.

Level accumulation cancellation mode accumulates input signal levels and regards that the input level has reached 0 or 1 when the accumulation result becomes 0 or the specified value.

Each period is specified by the noise cancellation register in each channel, and the noise cancellation counter measures the time.

Figure 21.1 and **Figure 21.3** show schematics of noise cancellation operation (example of TIA00 input of timer A) in premature-transition cancellation mode and level accumulation cancellation mode, respectively.

Edge detection is made for signals that have passed through the noise canceler in each channel. **Figure 21.1** and **Figure 21.3** show examples of rising-edge detection for signals that have passed through the noise canceler.

(1) NCM2C7 — Subblock 7 Noise Cancellation Mode 2

This bit specifies the operating mode of subblock 7 noise canceler.

(2) NCM2C6 — Subblock 6 Noise Cancellation Mode 2

This bit specifies the operating mode of subblock 6 noise canceler.

(3) NCM2C5 — Subblock 5 Noise Cancellation Mode 2

This bit specifies the operating mode of subblock 5 noise canceler.

(4) NCM2C4 — Subblock 4 Noise Cancellation Mode 2

This bit specifies the operating mode of subblock 4 noise canceler.

(5) NCM2C3 — Subblock 3 Noise Cancellation Mode 2

This bit specifies the operating mode of subblock 3 noise canceler.

(6) NCM2C2 — Subblock 2 Noise Cancellation Mode 2

This bit specifies the operating mode of subblock 2 noise canceler.

(7) NCM2C1 — Subblock 1 Noise Cancellation Mode 2

This bit specifies the operating mode of subblock 1 noise canceler.

(8) NCM2C0 — Subblock 0 Noise Cancellation Mode 2

This bit specifies the operating mode of subblock 0 noise canceler.

CAUTIONS

1. **Setting this register is enabled only when the NCMC bit in the noise cancellation mode register (NCMR) of the common controller is 0 and the corresponding channel bit (NCM1Cx) in the noise cancellation mode channel register 1C (NCMCR1C) of timer C is 1.**
2. **x = 0 to 7**

Table 21.80 shows the truth table for settings that determine noise cancellation modes.

Table 21.80 Truth Table for Noise Cancellation Modes of Timer C

Channel enable	Filter mode		Channel register			
NCCRCx. NCECx	NCMR. NCMC	NCMR. NCMSEL	NCMCR1C. NCM1Cx	NCMCR2C. NCM2Cx	Operating mode	Unit of filtering
0	—	—	—	—	Filter invalid	—
1	0	(—)	0	(—)	Premature-transition	Each subblock/ channel
1	0	(—)	1	0	Minimum time-at-level	Each channel
1	0	(—)	1	1	Level accumulation	Each channel
1	1	(0)	—	—	Minimum time-at-level	Each subblock
1	1	1	—	—	Level accumulation	Each subblock

Note 1. x = 0 to 7

Note 2. An enable bit (NCECx) can be set for each channel.

Setting when each channel is the unit of filtering of timer C (precondition: channel enable = 1)

- Setting each channel

  When the noise cancellation mode bit (NCMC) in the noise cancellation mode register (NCMR) is set to 0 and any of the noise cancellation mode channel setting bits (NCM1C*) in the noise cancellation mode channel register 1C (NCMCR1C) of timer C is set to 1

- Setting all channels

  When the noise cancellation mode bit (NCMC) in the noise cancellation mode register (NCMR) is 1 in the description of “Setting each channel” above or when all of the noise cancellation mode channel setting bits (NCM1Cx) in the noise cancellation mode channel register 1C (NCMCR1C) are set to 0 in the description of “Setting each channel” above

21.6.3 Operation

21.6.3.1 Input Capture Mode — Input Capture Function

Timer general registers Cxy (GRCxy) of timer C performs input capture when an edge is input from the corresponding external pin (TIOCxy) or an event output (event 1, 2A, or 2B) from timer A is input by setting input capture in timer I/O control registers Cx (TIORCx).

The timer counter Cx (TCNTCx) starts up-counting by setting the timer start register C (TSTRC). When an edge of the corresponding external pin of GRCxy is input or an event output from timer A is input, the corresponding bit (IMFCxy) in the timer status register Cx (TSRCx) is set to 1 and the counter value is transferred to GRCxy. After a changed edge of TIOCxy has been extracted and then two internal operating clock (PCLK) cycles have passed, the interrupt output changes. Input capture flags (IMFCxy bits in TSRCx) can be cleared by writing 1 to the corresponding clear bit in the timer status clear register Cx (TSCRCx).

An edge to be input can be selected from rising edge, falling edge, and both edges according to the setting of the IOC bit in the TIORCx register.

Furthermore, an interrupt request can be output to the CPU at the input capture timing and DMA transfer can be activated.

The output compare register Cxy (OCRCxy) and timer counter Cx (TCNTCx) are compared.

When the values in OCRCxy and TCNTCx match, the bit (OCMFCxy) in timer status register Cx (TSRCx) corresponding to OCRCxy is set.

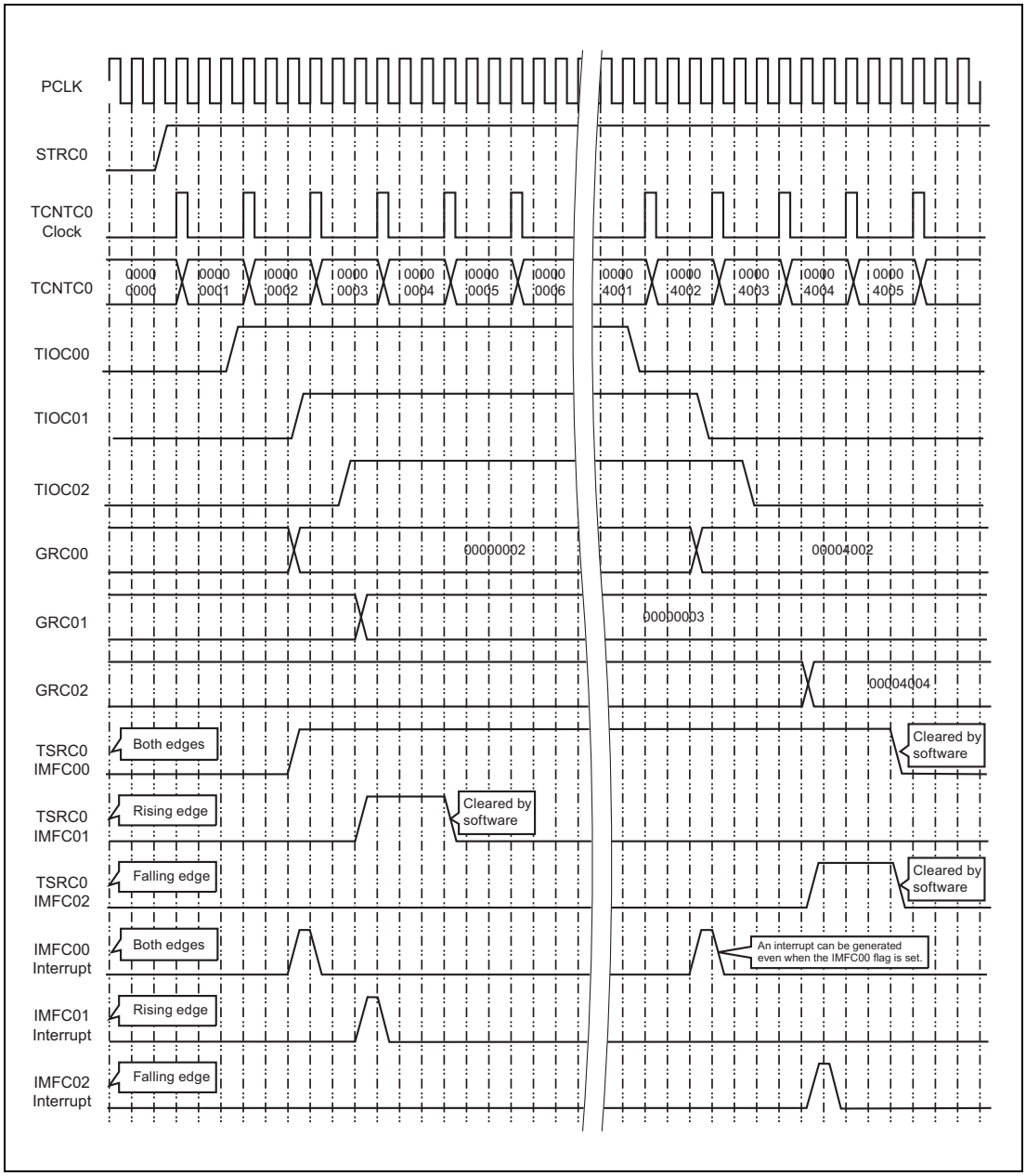


Figure 21.32 Operation Example of Input Capture

Figure 21.32 shows an operation example of input capture in subblock C0. In this example, the both edges on pin TIOC00, the rising edge on pin TIOC01, and the falling edge on TIOC02 are selected.

Input capture is performed even if TCNTCx is stopped (when the TCE bit in ATUENR or bit STRCx is cleared to 0). The counter value of the TCNTCx stopped is captured in GRCxy.

21.6.3.2 Compare Match Mode — Compare Match Function

Timer general registers Cxy (GRCxy) of timer C output compare match from the corresponding external pin (TIOCxy) by setting compare match operation in timer I/O control registers Cx (TIORCx).

Timer counter Cx (TCNTCx) is started for counting up by setting a bit in timer start register C (TSTRC) to 1. Set the value in GRCxy before starting the counter. When the values in GRCxy and TCNTCn match, a bit corresponding to GRCxy in timer status register Cx (TSRCx) is set and a waveform is output on external pin TIOCxy.

The compare match flag is set and the signal level on pin TIOCnm is changed on the first edge of the PCLK immediately after compare match between GRCnm and TCNTCx. Compare match flags (IMFCxy bits in TSRCx) can be cleared to 0 by writing 1 to the corresponding clear bit in the timer status clear register Cx (TSCRCx).

A logical one, a logical zero, or a toggled output can be selected for the signal to be output on pin TIOCxy.

An interrupt request can be output to the CPU at occurrence of a compare match.

and DMA transfer can be activated.

The output compare register Cxy(OCRCxy) is compared with the timer counter Cx(TCNTCx). When the values of OCRCxy and TCNTCx are matched, the (OCMFCxy) bit in the timer status register Cx (TSRCx) corresponding to OCRCxy is set.

Figure 21.33 shows an operation example of compare match. In this example, a toggled output on GRC00, a logical one output on GRC01, and a logical zero output on GRC02 are externally output. A value of 004004 is set for GRC0y, which changes the TIOC0y output on the next PCLK after compare match with the TCNTC0 value.

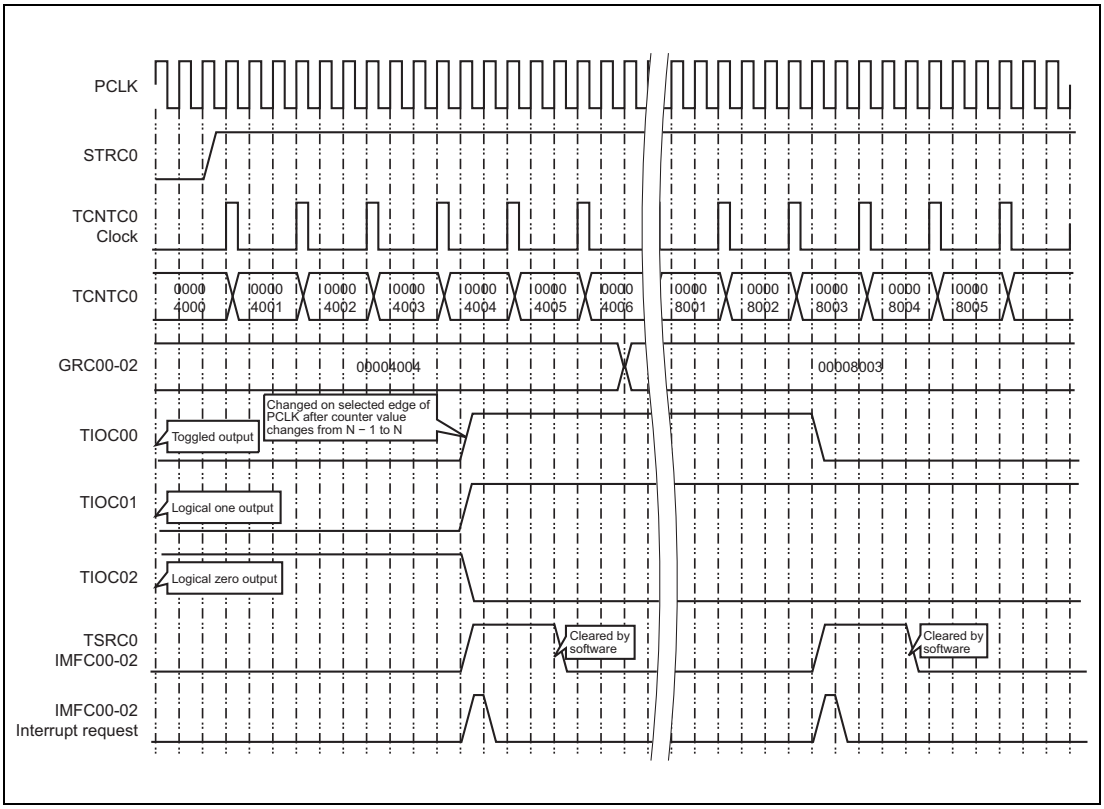


Figure 21.33 Operation Example of Compare Match

By setting a forced compare match bit (FMCMxy) in timer control register Cx (TCRCx), compare match can be generated even if TCNTCx has not matched GRCxy. The compare match flag and the signal level on pin TIOCxy are changed on the first edge of the PCLK after bit FCMCxy is set to 1.

The match in comparison is detected when any of the following occurs.

- When the values in TCNTCx and GRCxy match (forced compare match is OFF)
- When the forced compare match bit (bit FCMCxy in TCRCx) is changed from 0 to 1
- When the values in TCNTCx and GRCxy match because a counter is cleared on compare match of GRCx0

The compare match flag and the signal level on pin TIOCxy is changed on the first edge of the PCLK after bit FCMCxy is set to 1. Make sure that the compare match operation is selected by bits IOCxy[2:0] in TIORCx before starting operation. The match in comparison is not detected; when GRCxy is set to the same value as TCNTCx; when the compare match operation is selected after setting the forced compare match bit to 1.

The match in comparison is detected regardless of the counter operating state. Even if the counter is stopped, compare match occurs when the condition is satisfied.

When the compare match status flag is cleared before GRCxy and TCNTCx are changed (such as before counting while the counter has been stopped), the match in comparison is not detected.

The output compare register Cxy (OCRCxy) and timer counter Cx (TCNTCx) are compared.

When the values in OCRCxy and TCNTCx match, the bit (OCMFCxy) in timer status register Cx (TSRCx) corresponding to OCRCxy is set.

21.6.3.3 PWM Function

Setting bit PWMx0 in timer control register Cx (TCRCx) to 1 makes channels 1 to 3 in each subblock function as PWM timers with the same frequency. In PWM mode, GRCx0 as a cycle setting register and GRCx1 to GRCx3 as duty cycle setting registers are used. External pins TIOCx1 to TIOCx3 corresponding GRCx1 to GRCx3 are used to output PWM signals. To use them as PWM signal outputs, select the compare match operation by bits IOCxy in TIORCx as well as setting bit PWMx0 so that GRCn0 to GRCn3 function as compare match registers. Furthermore, the active polarity of the PWM waveform can be set by the IOCxy[0] bit in TIORCx (IOCxy[0] = 0: Active high, 1: Active low)

Figure 21.34 shows the PWM operation setting procedure

Timer counter Cx (TCNTCx) is started by setting timer start register C (TSTRC). When the value in TCNTCx reaches the value in the cycle setting register (GRCx0), compare match occurs and the bit in timer status register Cx (TSRCx) is set to 1. At this time, TCNTCx is cleared and a signal is output on external pins TIOCx1 to TIOCx3 according to bit IOCx0 in PWM mode. The output signal levels on pin TIOCx0 depend on bit IOCx0.

When the value in TCNTCx reaches the value in the duty cycle setting register (GRCx1 to GRCx3), the bit in timer status register Cx (TSRCx) is set to 1 and a signal is output on external pins TIOCx1 to TIOCx3 according to bits IOCx1 to IOCx3.

When the same value is set in the cycle and duty cycle setting registers, priority is given to bit IOCx0 corresponding to the cycle setting register.

Figure 21.35 shows an operation example of block C0 in PWM mode.

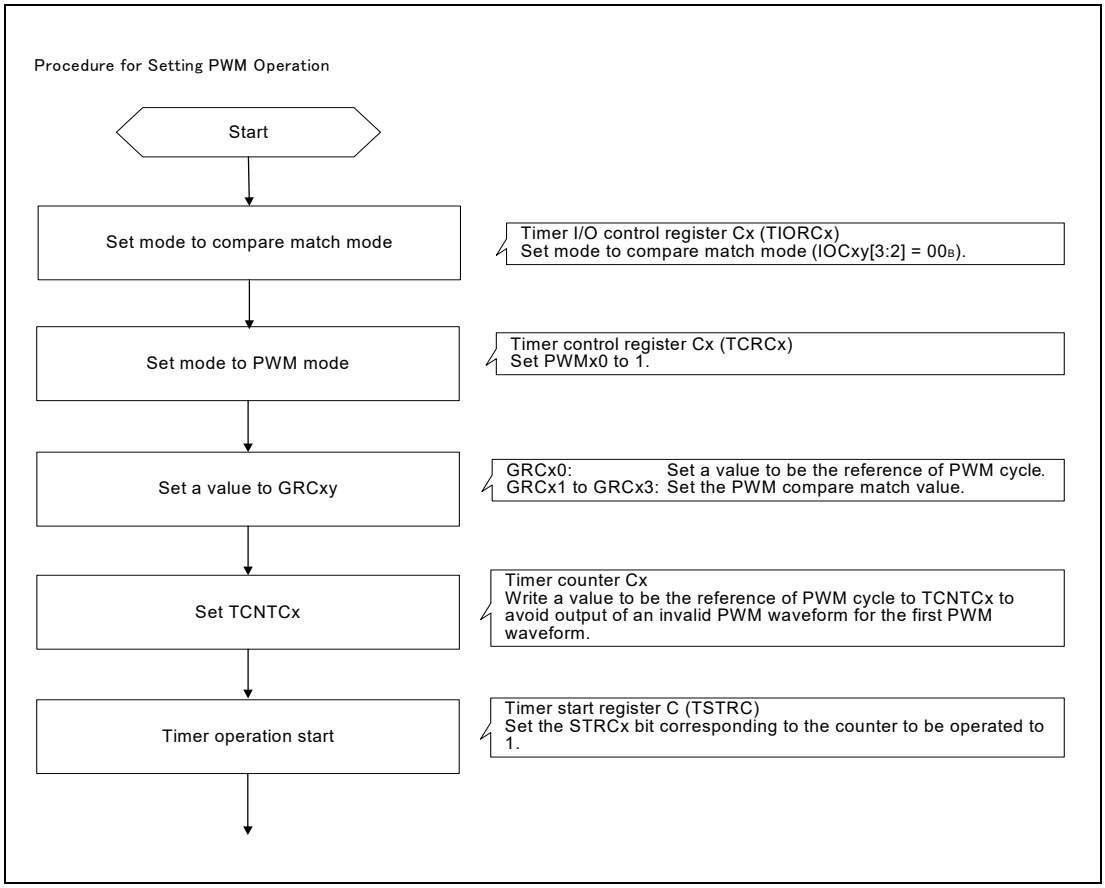


Figure 21.34 PWM Operation Setting Procedure

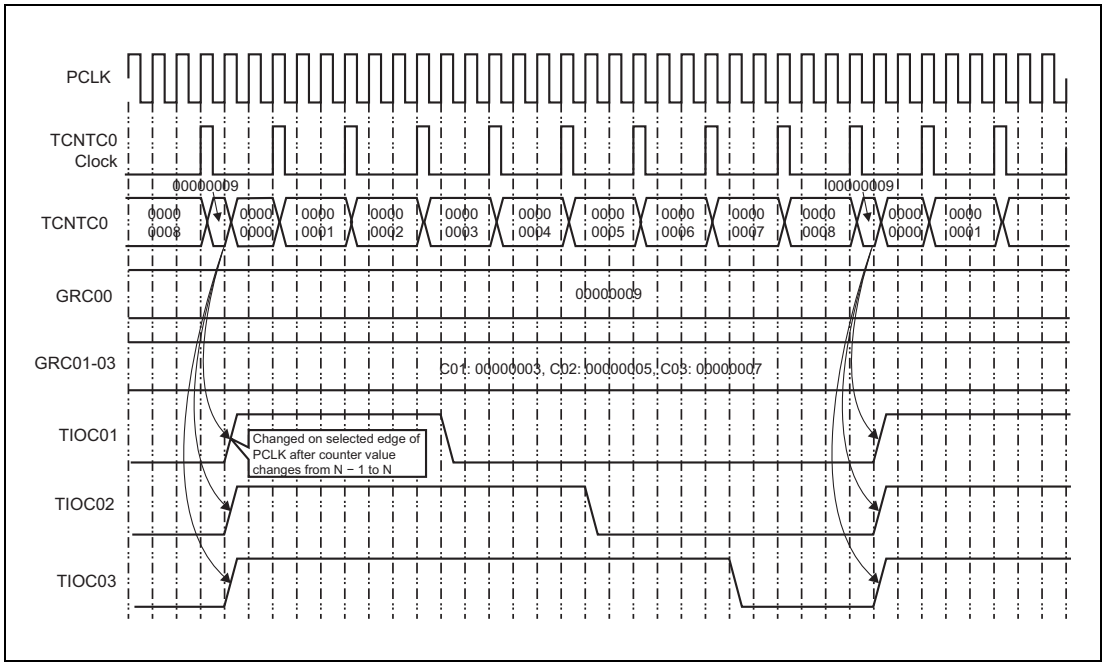


Figure 21.35 Operation Example of PWM Mode

21.6.3.4 One-Shot Pulse Mode — One-Shot Pulse Function

By setting the PWMx0 bits in timer control register Cx (TCRCx) to 0_B and setting the IOCxy[3:2] bits in timer I/O control register Cx (TIORXCx) to 10_B, one-shot pulses are output from the corresponding external pin (TIOCxy).

Setting the timer start register C (TSTRC) starts up-count of the timer counter Cx (TCNTCx).

In one-shot pulse mode, a one-shot pulse is output at a trigger of write access to the GRCxy register.

After the write access to the GRCxy register, a compare match interrupt is output on compare match between timer counter Cx (TCNTCx) and OCRCxy register, set the external pin (TIOCxy) to active level, a compare match interrupt is output on compare match between timer counter Cx (TCNTCx) and GRCxy register, and then reset the external pin (TIOCxy) to inactive level. When a single one-shot pulse output has been completed after an access to the GCRxy register, no interrupt occurs until the next write access to the GCRxy register is made and the external pin (TIOCxy) retains inactive level.

When one-shot pulse mode (IOCxy[3:2] = 10_B) is used, initialize the external pin (TIOCxy) by forced compare match Cxy (FCMCxy) after operating mode is set, and then start counting.

If one-shot pulse mode is activated without initializing the external pin (TIOCxy) to inactive level, output level becomes unstable.

To cancel one-shot pulse mode after a value is written to the GRCxy register, write 1 to forced compare match Cxy (FCMCxy) and forced output compare match Cxy (FOCMCxy) simultaneously.

Figure 21.36 shows an example of one-shot pulse mode operation in block C0.

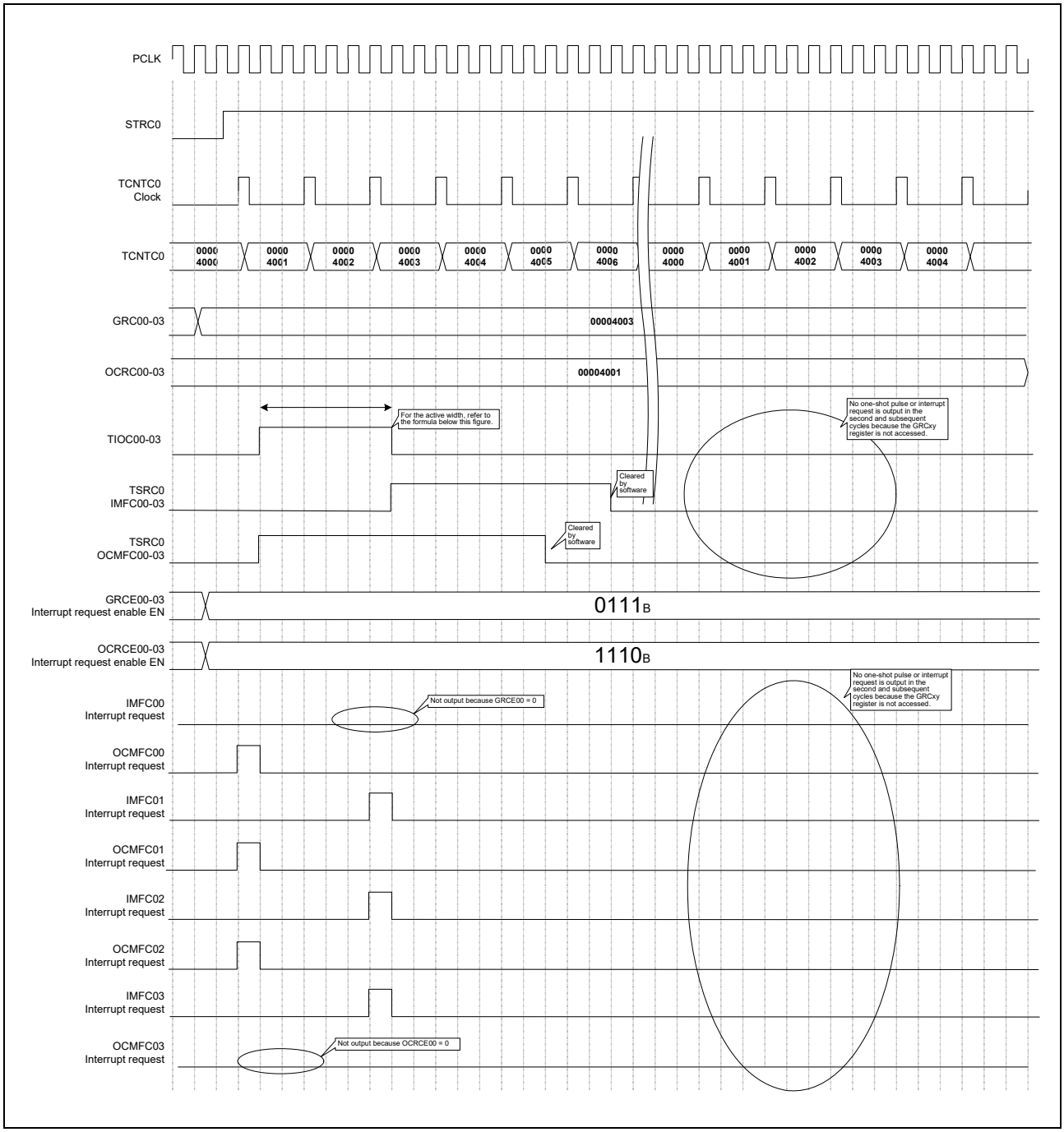


Figure 21.36 One-Shot Pulse Mode Operation

The active width of one-shot pulse can be obtained from the following formula.

GRCxy = OCRCxy:

One-shot pulse (active) width = 0

GRCxy > OCRCxy:

One-shot pulse (active) width = GRCxy[31:0] – OCRCxy[31:0]

GRCxy < OCRCxy:

- Upper-limit setting function is off

  One-shot pulse (active) width = (FFFF $FFFF_H$ – OCRCxy[31:0]) + (GRCxy[31:0] + 1)

- Upper-limit setting function is on

  One-shot pulse (active) width = (CUCRCx – OCRCxy[31:0]) + GRCxy[31:0]

IMFC and OCMFC shares one interrupt request signal though described separately in the above figure.

To output a one-shot pulse in the second and subsequent cycles, be sure to confirm that the previously set one-shot pulse output has been completed, and then set GRCxy and OCRCxy in the second and subsequent cycles. If the GRCxy or OCRCxy register value is modified before the one-shot pulse output has been completed, the modified value is reflected to the ongoing one-shot-pulse output.

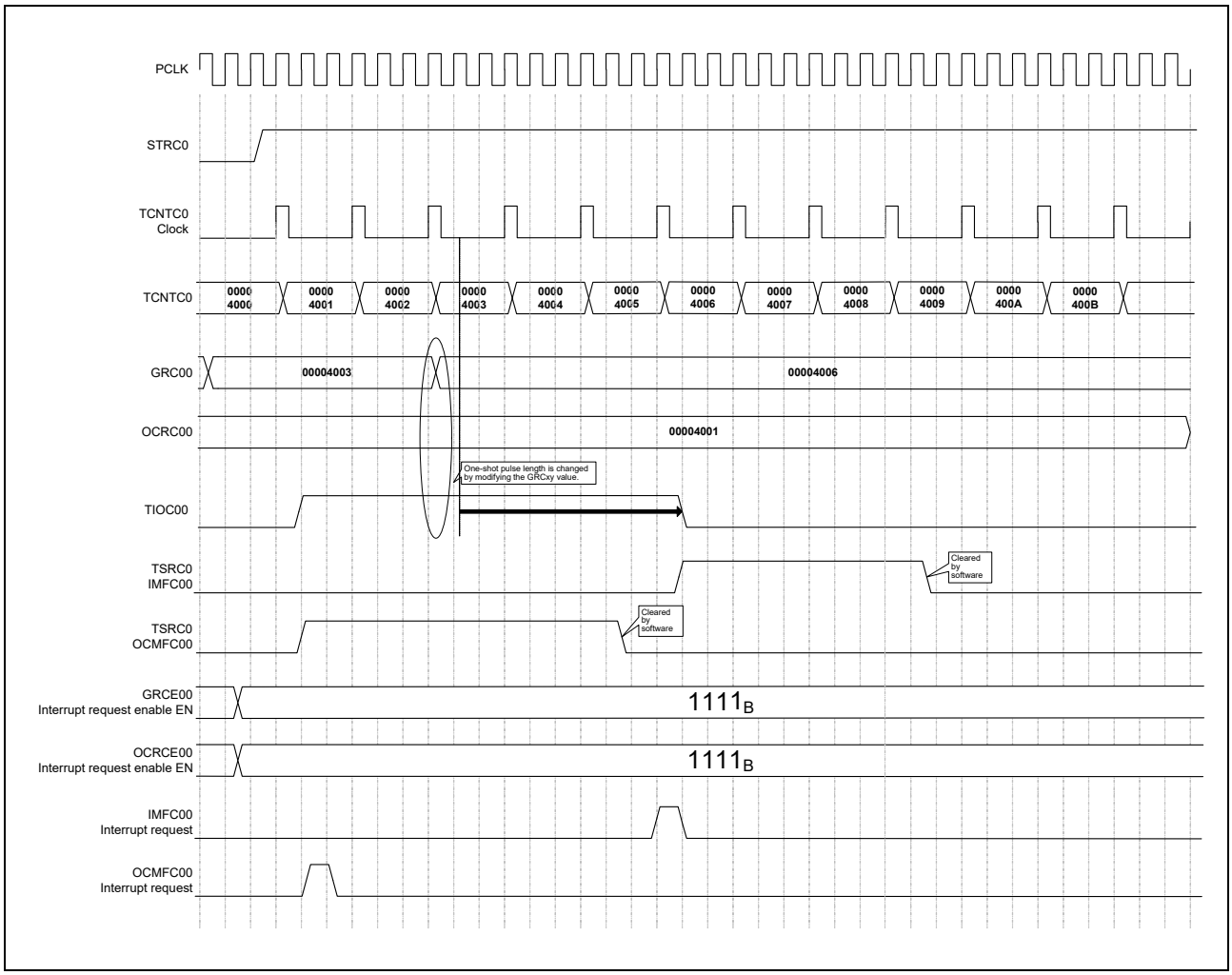


Figure 21.37 Operation when the Compare Register Is Modified during One-Shot Pulse Output

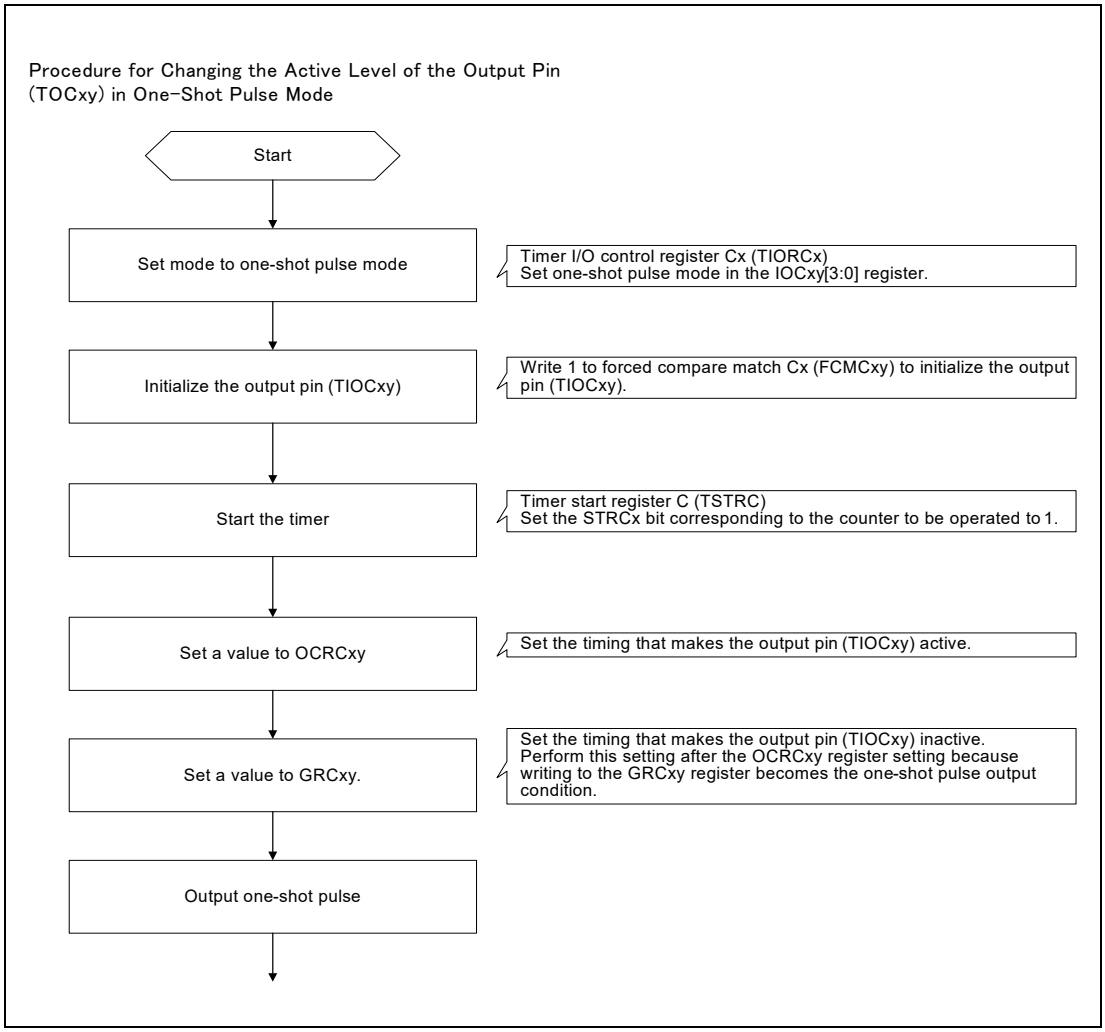


Figure 21.38 One-Shot Pulse Operation (Output Pin Initialization Procedure)

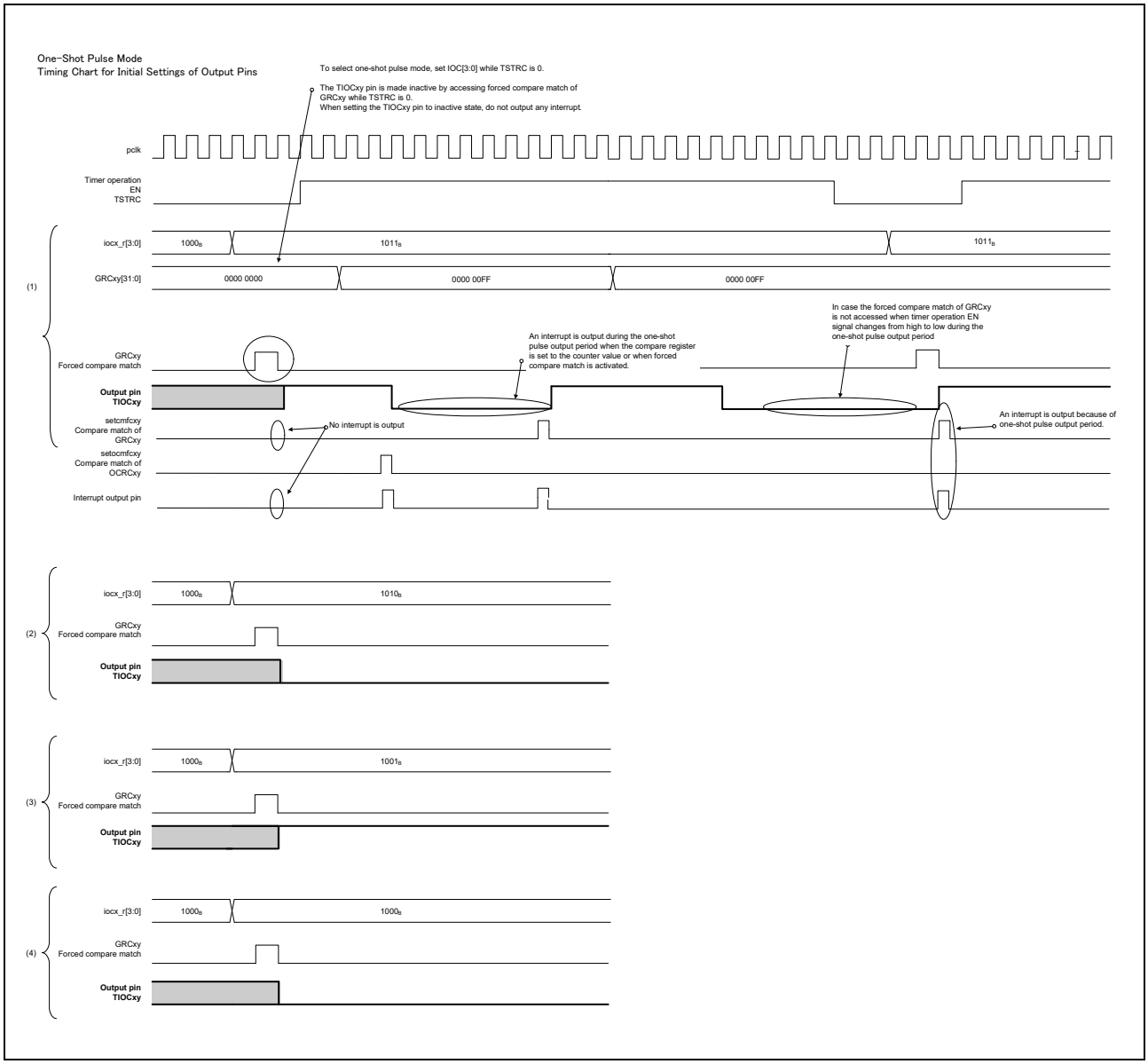


Figure 21.39 One-Shot Pulse Operation (Output Pin Initialization Timing Chart)

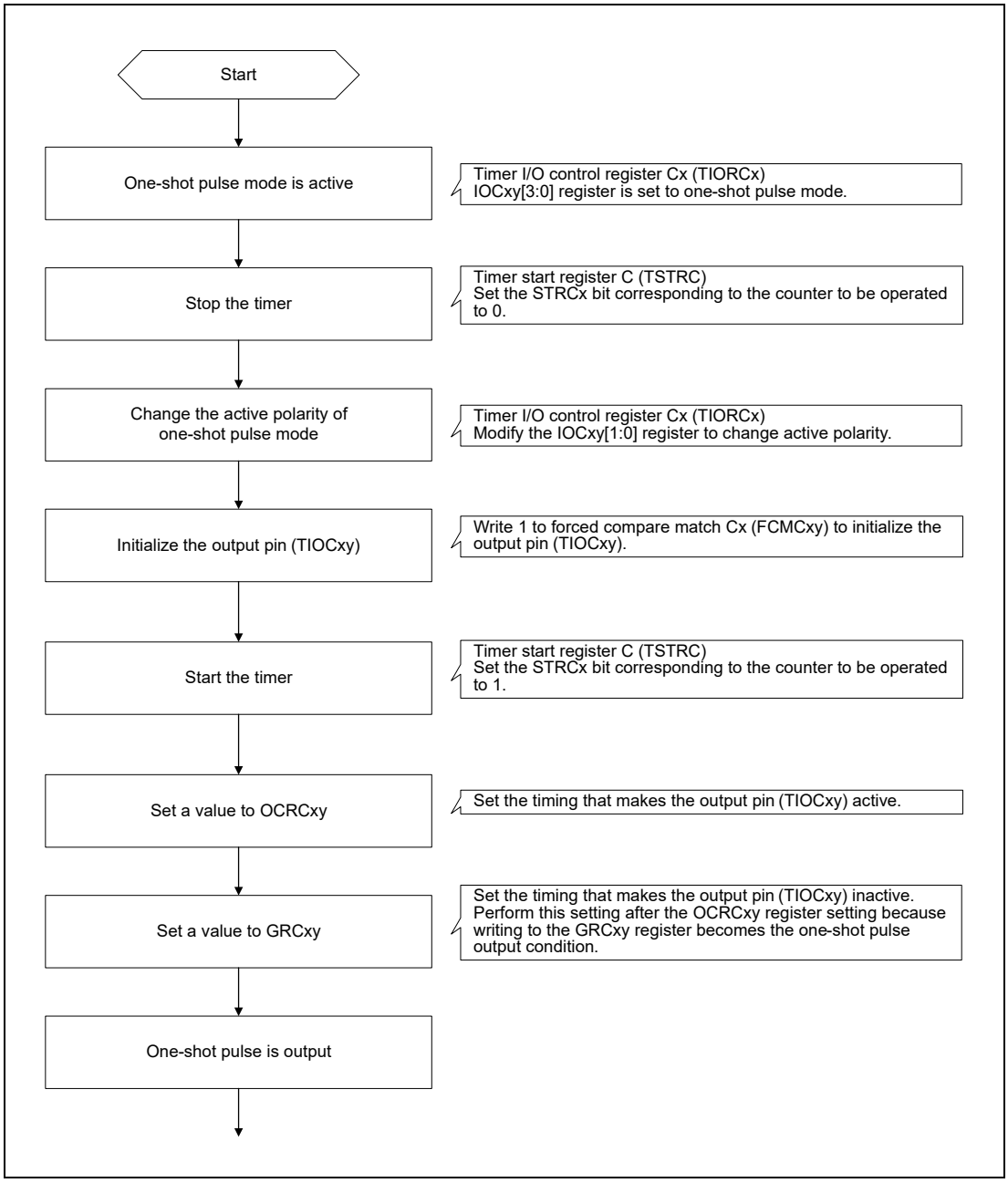


Figure 21.40 One-Shot Pulse Operation (Outline of Active Polarity Change)

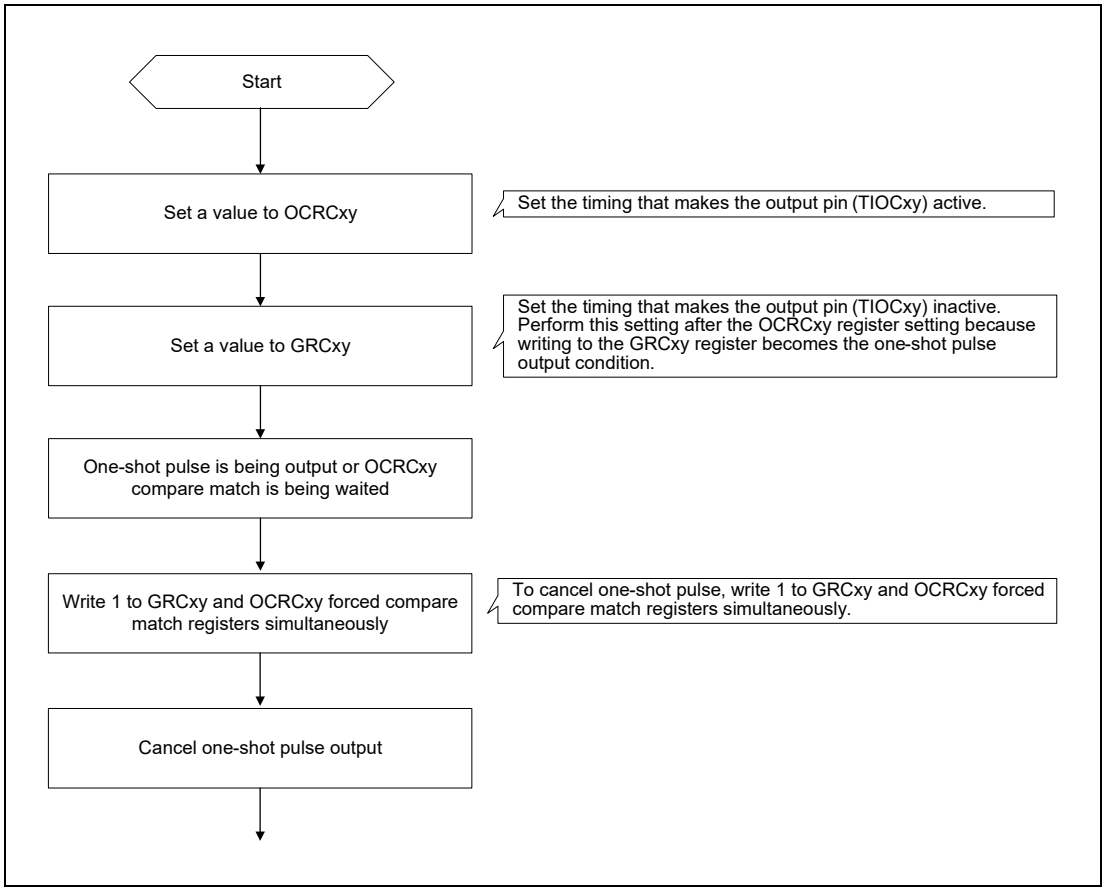


Figure 21.41 One-Shot Pulse Output Cancellation Procedure

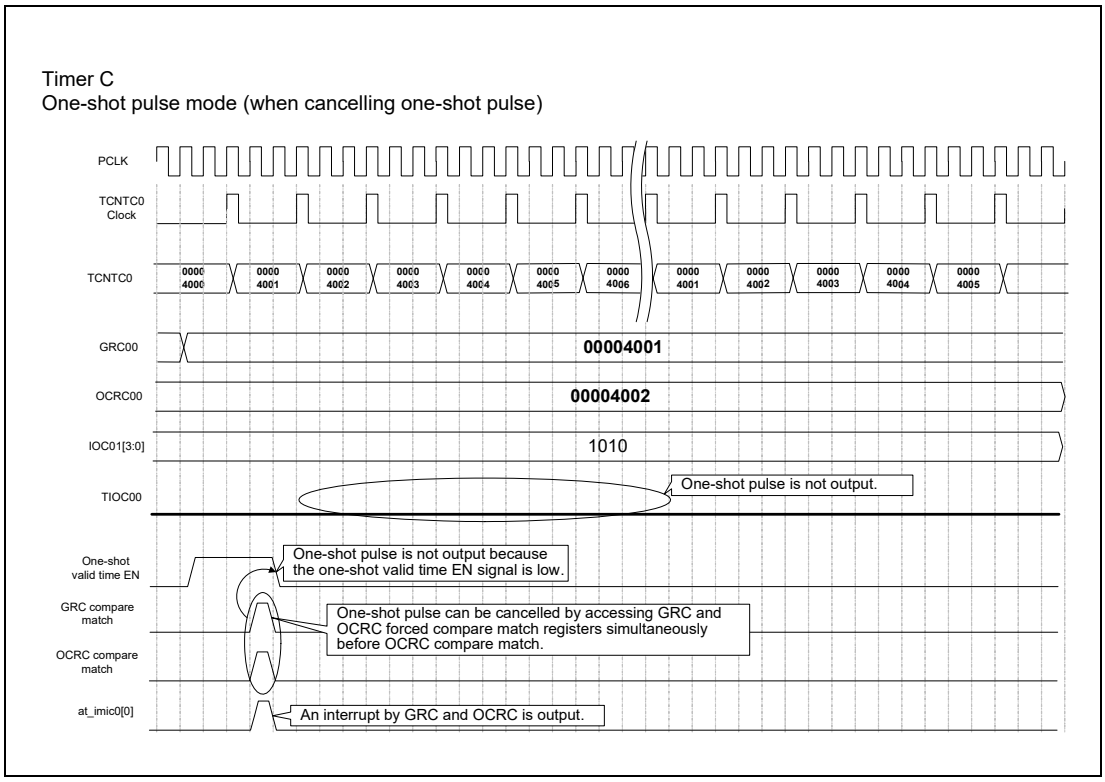


Figure 21.42 One-Shot Pulse Output Cancellation Timing Chart

21.6.3.5 Counter Upper-Limit Setting Function

The timer counter Cx (TCNTCx) upper-limit value in each subblock can be changed by the counter upper-limit setting compare register Cx (CUCRCx) to perform interval operation. To enable the counter upper-limit setting function, set the CLRCx bit in the timer control register Cx (TCRCx) to 1.

Figure 21.43 shows an example of operation with the counter upper-limit setting function of block C0 enabled.

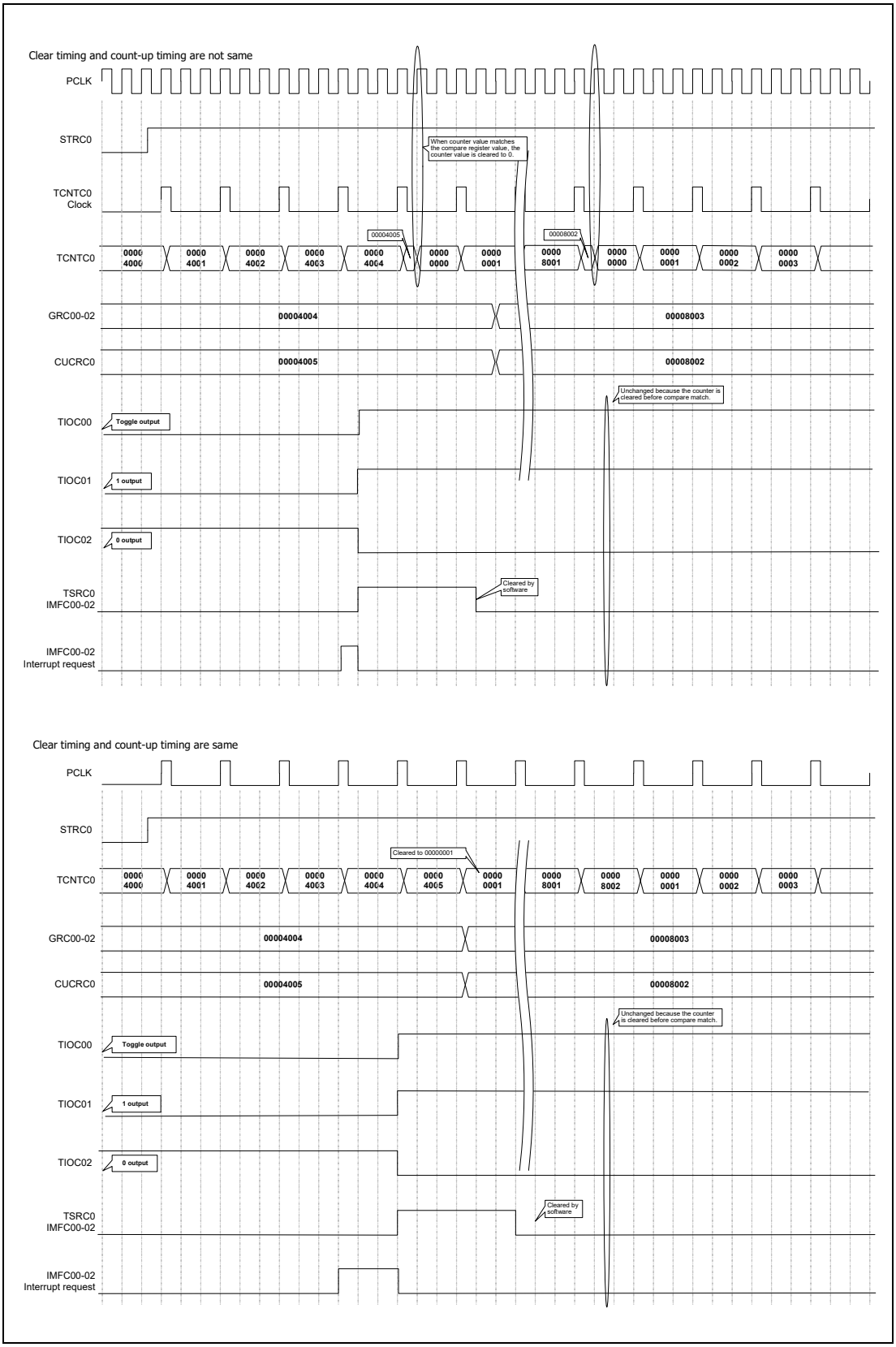


Figure 21.43 Example of Upper-Limit Setting Function Operation

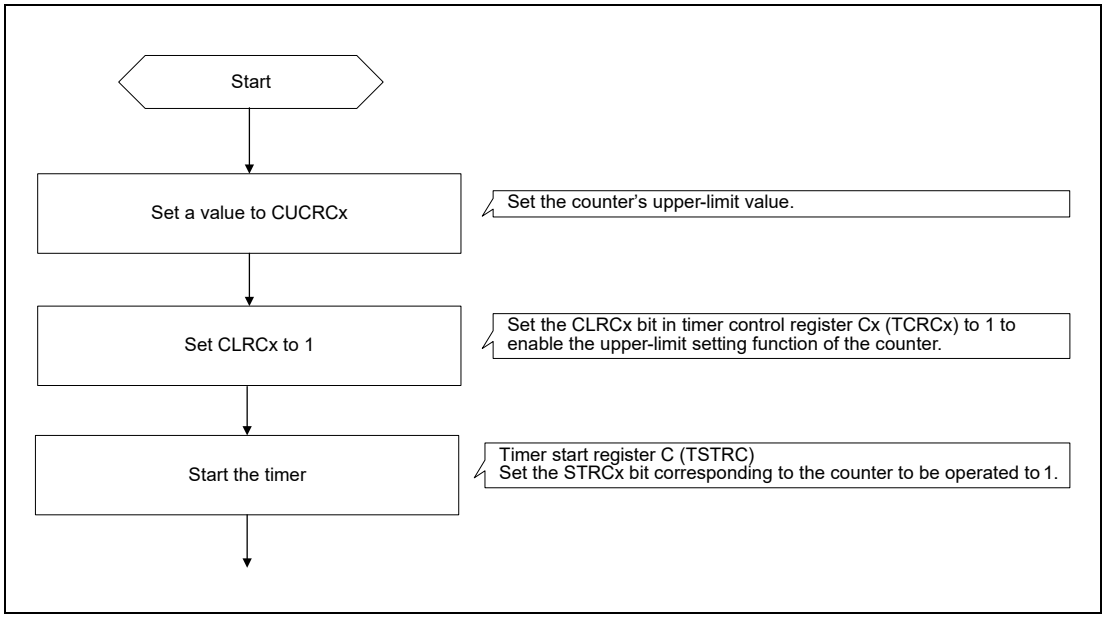


Figure 21.44 Counter Upper-Limit Setting Function Setting Procedure

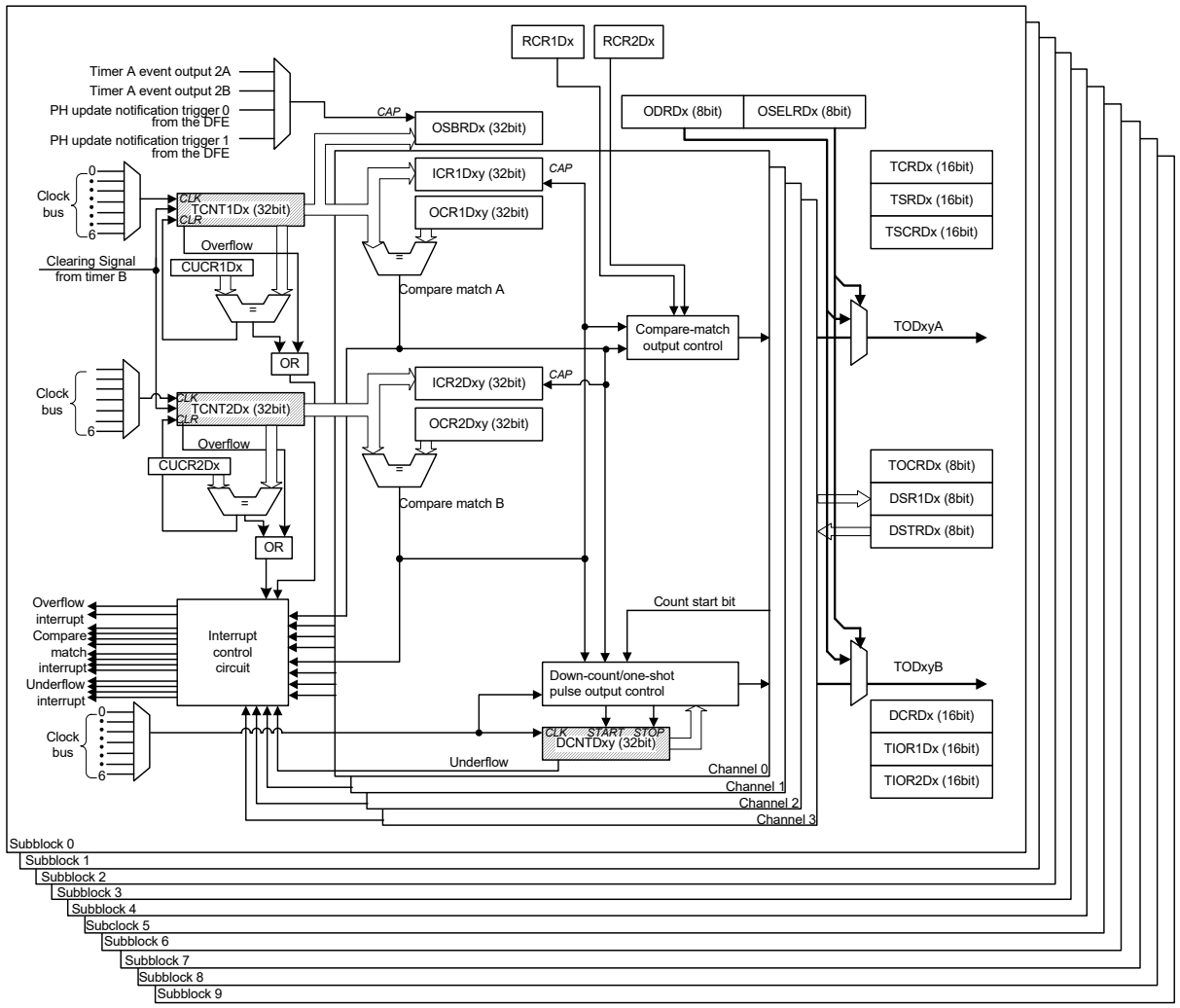
21.7 Timer D

21.7.1 Operation Overview

Timer D consists of ten subblocks that output one-shot pulses.

Timer D has the following functions.

- A compare match between the counter and output compare register is generated.
  Compare match A by compare match of TCNT1Dx and OCR1Dxy and compare match B by compare match of TCNT2Dx and OCR2Dxy are generated respectively.
- Down counter is started on compare match A or B of the output compare register or writing to the counter start bit in the down-counter start register. A one-shot pulse with an offset can be output.
- The output waveform can be forcibly shutoff regardless of the down-counter value by compare match B of the compare match register. If compare match A or counter start bit writing occurs concurrently with compare match B, output shutoff takes precedence.
- The input capture register enables to capture the TCNT2Dx value using compare match A as a trigger and the TCNT1Dx value using compare match B as a trigger.
- Pulse indicating that compare match A or B has been detected for A/D converter activation can be output. (40 lines supported by subblocks D0, D2, D4, D6, and D9)
- Interrupt requests can be output on compare matches A and B (80 lines supported).
- Range compare function by the compare match register is enabled according to the range comparison value setting register.
- When the upper-limit setting function of the timer counter 1Dx (TCNT1Dx) is enabled, the counter can be cleared on compare match of CUCR1Dx.
- When the upper-limit setting function of the timer counter 2Dx (TCNT2Dx) is enabled, the counter can be cleared on compare match of CUCR2Dx.
- Interrupt requests (20 in total) can be output on counter overflow: ten outputs from TCNT1Dx and ten outputs from TCNT2Dx.
- Interrupt requests can be output on down-counter underflow (40 lines supported). A DMA transfer request can be issued for the DMAC. (20 lines for DMA transfer requests supported by down counters in subblocks D0 to D4)
- Offset base register can capture the counter value by a trigger signal from timer A and a PH notification trigger from DFE
- Clearing TCNT1Dx and TCNT2Dx is possible by a clear signal issued from timer B.
- The value configured in the output value register can be output from pins according to the output setting register.
- Reference data and events can be output to APA. The counter values of TCNT1D0 and TCNT2D0 can be output as reference data. For selection of output values, see **Section 22.5.8, APA Input Selector**. Compare match A, compare match B, down counter underflow, TOD0yA, and TOD0yB of subblock 0 can be output as event outputs.



Note: x = 0 to 9 (number of Subblock), y = 0 to 3 (number of channel)

Figure 21.45 Block Diagram of Timer D

Figure 21.45 is a block diagram of timer D. Components of timer D subblock Dx include two range compare setting registers (RCR1Dx and RCR2Dx), two 32-bit timer counters 1Dx, 2Dx (TCNT1Dx and TCNT2Dx), one offset base register Dx (OSBRDx), one output value register Dx (ORDx), one output setting register Dx (OSELRDx), eight output compare registers 1Dxy, 2Dxy (OCR1Dxy and OCR2Dxy), four timer down-counters Dxy (DCNTDxy), two counter upper-limit setting registers Dx (CUCRDx), eight input capture registers 1Dxy, 2Dxy (ICR1Dxy and ICR2Dxy), and a controller. Each channel includes two output pins; TODxyA for compare match output and TODxyB for one-shot pulse output. (Output pins are supported by subblocks D0 to D8.)

TODxyA and TODxyB are output a level of 0 as a default.

A trigger for activating the A/D converter can be output to the on compare matches A and B.

21.7.2 Registers Related to Timer D

21.7.2.1 TSTRD — Timer Start Register D

Access: 8-bit/16-bit accessible

Address: FFE6 1000_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	STRD9	STRD8	STRD7	STRD6	STRD5	STRD4	STRD3	STRD2	STRD1	STRD0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.81 TSTRD Register Contents

Bit Position	Bit Name	Function
15 to 10	—	Not used. This bit is always read as 0. These bits are read as 0. When writing, always write 0.
9	STRD9	Counter D9 Start 0: TCNT1D9, TCNT2D9, and DCNTD9y are disabled 1: TCNT1D9, TCNT2D9, and DCNTD9y are enabled
8	STRD8	Counter D8 Start 0: TCNT1D8, TCNT2D8, and DCNTD8y are disabled 1: TCNT1D8, TCNT2D8, and DCNTD8y are enabled
7	STRD7	Counter D7 Start 0: TCNT1D7, TCNT2D7, and DCNTD7y are disabled 1: TCNT1D7, TCNT2D7, and DCNTD7y are enabled
6	STRD6	Counter D6 Start 0: TCNT1D6, TCNT2D6, and DCNTD6y are disabled 1: TCNT1D6, TCNT2D6, and DCNTD6y are enabled
5	STRD5	Counter D5 Start 0: TCNT1D5, TCNT2D5, and DCNTD5y are disabled 1: TCNT1D5, TCNT2D5, and DCNTD5y are enabled
4	STRD4	Counter D4 Start 0: TCNT1D4, TCNT2D4, and DCNTD4y are disabled 1: TCNT1D4, TCNT2D4, and DCNTD4y are enabled
3	STRD3	Counter D3 Start 0: TCNT1D3, TCNT2D3, and DCNTD3y are disabled 1: TCNT1D3, TCNT2D3, and DCNTD3y are enabled
2	STRD2	Counter D2 Start 0: TCNT1D2, TCNT2D2, and DCNTD2y are disabled 1: TCNT1D2, TCNT2D2, and DCNTD2y are enabled
1	STRD1	Counter D1 Start 0: TCNT1D1, TCNT2D1, and DCNTD1y are disabled 1: TCNT1D1, TCNT2D1, and DCNTD1y are enabled
0	STRD0	Counter D0 Start 0: TCNT1D0, TCNT2D0, and DCNTD0y are disabled 1: TCNT1D0, TCNT2D0, and DCNTD0y are enabled

Note: A start bit is provided for each subblock D0 to D9. (y = 0, 1, 2, 3: Correspond to Dx0 to Dx3.)

Timer start register D (TSTRD) is a 16-bit register that can read and write upper 8 bits as TSTRDH and lower 8 bits as TSTRDL. This register is set to enable or disable two timer counters 1Dx and 2Dx (TCNT1Dx and TCNT2Dx) and timer down counters Dxy (DCNTDxy) in nine subblocks Dx (x = 0 to

9). When the count Dx start bit and the TDE bit in the ATU-IV control register (ATUENR) are both set to 1, the counters are started.

When TSTRD is reset, it is initialized to 0000_H.

(1) STRDx — Counter Dx Start Bit

These bits enable and disable timer counters 1Dx and 2Dx (TCNT1Dx, TCNT2Dx) and timer down counters Dxy (DCNTDxy).

When this bit is cleared to 0, TCNT1Dx, TCNT2Dx, and DCNTDxy stop operation. The counter value is retained while the counter is stopped. When this bit is set to 1 again, the counter is restarted from the value.

Even if the counter Dx start bit is set to 1, counting does not start unless the master enable of timer Dx is set to 1 by the ATU-IV control register.

CAUTION

The prescalers run regardless of this counter Dx start bit and are not synchronized with the timing at which this bit is set. Therefore, the time from when this bit is set to when TCNT1Dx and TCNT2Dx are incremented for the first time is less than the cycle of the clock of TCNT1Dx and TCNT2Dx.

21.7.2.2 TCRDx — Timer Control Registers Dx

(x = 0, 1, 2, 3, 4, 5, 6, 7, 8, 9: Corresponding to subblocks D0 to D9, y = 0, 1, 2, 3: Corresponding to channels 0 to 3)

Access: 8-bit and 16-bit accessible

Address: FFE6 1100_H + (100_H * x)

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	OBRE Dx	C2CE Dx	C1CE Dx	CLR2 Dx	CKSEL2Dx[2:0]			CLR1 Dx	CKSEL1Dx[2:0]			—	DCSELDx[2:0]		
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W

Table 21.82 TCRDx Register Contents

Bit Position	Bit Name	Function
15	—	Not used. This bit is always read as 0. This bit is always read as 0. When writing, always write 0.
14	OBREDx	Timer Offset Base Register Enable Enables and disables input capture to the offset base register.
13	C2CEDx	Counter 2 Clear Enable Enables and disables a timer counter 2 clearing request from timer B.
12	C1CEDx	Counter 1 Clear Enable Enables and disables a timer counter 1 clearing request from timer B.
11	CLR2Dx	TCNT2Dx Clear Setting Enables and disables clearing the timer counter.
10 to 8	CKSEL2Dx[2:0]	TCNT2Dx Clock Select These bits select the TCNT2Dx counting-up clock from clock-bus lines 0 to 6.
7	CLR1Dx	TCNT1Dx Clear Setting Enables and disables clearing the timer counter.
6 to 4	CKSEL1Dx[2:0]	TCNT1Dx Clock Select These bits select the TCNT1Dx counting-up clock from clock-bus lines 0 to 6.
3	—	Not used. This bit is always read as 0. When writing, always write 0.
2 to 0	DCSELDx[2:0]	DCNTDxy Clock Select These bits select the DCNTDxy counting-down clock from clock-bus lines 0 to 6.

Timer control registers Dx (TCRDx) are 16-bit readable/writable registers that select the counter clocks in subblock Dx for 32-bit timer counter 1Dx (TCNT1Dx), 32-bit timer counter 2Dx (TCNT2Dx), and 32-bit timer down counter Dxy (DCNTDxy) from clock-bus lines 0 to 6. These registers also enable and disable capture of the timer offset base register, counter clearing requests from timer B for TCNT1Dx and TCNT2Dx, and clearing timer counters 1Dx (TCNT1Dx) and 2Dx (TCNT2Dx).

(1) OBREDx — Timer Offset Base Register Enable

This register enables and disables input capture to timer offset base register Dx (OSBRDx). When this bit is set to 1, the value in TCNT1Dx is captured by OSBRDx in the PCLK cycles following cycles in which the input signal selected by the TICTSELDx register is asserted. When the pulse width of the event 2A or 2B signal from timer A exceeds one cycle of PCLK, the counter value is captured on every clock cycle while the signal is at the active level.

OBREDx	Function	
0	Input capture by OSBRDx is enabled	(Initial value)
1	Input capture by OSBRDx is disabled	

(2) C2CEDx — Counter 2 Clear Enable

Enables and disables clearing the counter value in TCNT2Dn by timer B.

When an edge of the counter clearing signal output from timer B is detected while this bit is set to 1, TCNT2Dx is cleared ($0000\ 0000_H$) in the following timing. When the rising edge of the clearing signal is detected in the cycle in which TCNT2Dx is counted up, the counter is cleared on the counting up timing. When the rising edge of the clearing signal is detected in other than the TCNT2Dx counting-up cycle, the counter is cleared on the first counting up timing after edge detection.

The counter clearing signal is ignored with this bit set to the initial value.

C2CEDx	Function	
0	TCNT2Dx clearing signal from timer B is disabled	(Initial value)
1	TCNT2Dx clearing signal from timer B is enabled	

(3) C1CEDx — Counter 1 Clear Enable

Enables and disables clearing the value in TCNT1Dx by timer B.

When an edge of the counter clearing signal output from timer B is detected while this bit is set to 1, TCNT1Dx is cleared ($0000\ 0000_H$) in the following timing. When the rising edge of the clearing signal is detected in the cycle in which TCNT1Dx is counted up, the counter is cleared on the counting up timing. When the rising edge of the clearing signal is detected in other than the TCNT1Dx counting-up cycle, the counter is cleared on the first counting up timing after edge detection.

The counter clearing signal is ignored with this bit set to the initial value.

C1CEDx	Function	
0	TCNT1Dx clearing signal from timer B is disabled	(Initial value)
1	TCNT1Dx clearing signal from timer B is enabled	

(4) CLR2Dx — TCNT2Dx Clear

Setting the TCNT2Dx clear bit to 1 clears ($0000\ 0000_H$) the TCNT2Dx value at the first clock after a compare match between timer counter 2Dx (TCNT2D) and counter upper-limit setting compare register 2Dx (CUCR2Dx).

If there is a conflict with TCNT2Dx count up, it is cleared to $0000\ 0001_H$.

CLR2Dx	Function	
0	CUCR2Dx compare match is not used to clear the TCNT2Dx value.	(Initial value)
1	CUCR2Dx compare match is used to clear the TCNT2Dx value.	

When the CLR2Dx bit is set to 1, set the range comparison value setting register 2Dx (RCR2Dx) to 00_H.

(5) CKSEL2Dx[2:0] — TCNT2Dx Clock Select

These bits select the TCNT2Dx counting-up clock.

CKSEL2Dx				
[2]	[1]	[0]	Function	
0	0	0	Incrementation of TCNT2Dx is driven by clock-bus line 0.	(Initial value)
0	0	1	Incrementation of TCNT2Dx is driven by clock-bus line 1.	
0	1	0	Incrementation of TCNT2Dx is driven by clock-bus line 2.	
0	1	1	Incrementation of TCNT2Dx is driven by clock-bus line 3.	
1	0	0	Incrementation of TCNT2Dx is driven by clock-bus line 4.	
1	0	1	Incrementation of TCNT2Dx is driven by clock-bus line 5.	
1	1	0	Incrementation of TCNT2Dx is driven by clock-bus line 6.	
1	1	*	Reserved	

(6) CLR1Dx — TCNT1Dx Clear

The TCNT1Dx value is cleared to $0000\ 0000_H$ by setting the TCNT1Dx clear bit to 1 at the first clock after a compare match between timer counter 1Dx (TCNT1Dx) and counter upper-limit setting compare register 1Dx (CUCR1Dx). If there is a conflict with TCNT1Dx count up, it is cleared to $0000\ 0001_H$.

CLR1Dx	Function	
0	CUCR1Dx compare match is not used to clear the TCNT1Dx value.	(Initial value)
1	CUCR1Dx compare match is used to clear the TCNT1Dx value.	

When the CLR1Dx bit is set to 1, set the range comparison value setting register 1Dx (RCR1Dx) to 00_H.

(7) CKSEL1Dx[2:0] — TCNT1Dx Clock Select

These bits select the TCNT1Dn counting-up clock.

CKSEL1Dx				
[2]	[1]	[0]	Function	
0	0	0	Incrementation of TCNT1Dx is driven by clock-bus line 0.	(Initial value)
0	0	1	Incrementation of TCNT1Dx is driven by clock-bus line 1.	
0	1	0	Incrementation of TCNT1Dx is driven by clock-bus line 2.	
0	1	1	Incrementation of TCNT1Dx is driven by clock-bus line 3.	

CKSEL1Dx			
[2]	[1]	[0]	Function
1	0	0	Incrementation of TCNT1Dx is driven by clock-bus line 4.
1	0	1	Incrementation of TCNT1Dx is driven by clock-bus line 5.
1	1	0	Incrementation of TCNT1Dx is driven by clock-bus line 6.
1	1	*	Reserved

(8) DCSELDx[2:0] — DCNTDxy Clock Select

These bits select the clock signal to drive counting down by the 32-bit timer down counter Dxy (DCNTDxy) of the Dx sub-block.

DCSELDx				
[2]	[1]	[0]	Function	
0	0	0	Decrementation of DCNTDxy is driven by clock-bus line 0.	(Initial value)
0	0	1	Decrementation of DCNTDxy is driven by clock-bus line 1.	
0	1	0	Decrementation of DCNTDxy is driven by clock-bus line 2.	
0	1	1	Decrementation of DCNTDxy is driven by clock-bus line 3.	
1	0	0	Decrementation of DCNTDxy is driven by clock-bus line 4.	
1	0	1	Decrementation of DCNTDxy is driven by clock-bus line 5.	
1	1	0	Decrementation of DCNTDxy is driven by clock-bus line 6.	
1	1	*	Reserved	

Note: Channels 0 to 3 of a subblock share a common counter clock.

Output of one-shot pulse (TODxyB) is synchronized with the clock selected in these bits.

21.7.2.3 TIOR1Dx — Timer I/O Control Registers 1Dx

(x = 0, 1, 2, 3, 4, 5, 6, 7, 8, 9: Corresponding to subblocks D0 to D9, y = 0, 1, 2, 3: Corresponding to channels 0 to 3)

Access: 8-bit and 16-bit accessible

Address: FFE6 1104_H + (100_H * x)

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OSSDx3		OSSDx2		OSSDx1		OSSDx0		IOADx3		IOADx2		IOADx1		IOADx0	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.83 TIOR1Dx Register Contents

Bit Position	Bit Name	Function
15 to 8	OSSDx3 to OSSDx0	Compare Match Output Source Select These bits select the output level on the compare match output pin (TODxyA). The output is controlled by compare match A, compare match B, or both matches.
7 to 0	IOADx3 to IOADx0	I/O Control A These bits select the function of the output compare register 1Dxy (OCR1Dxy).

TIOR1Dx are 16-bit readable/writable registers that select the source of compare match output (TODxyA), enable and disable compare match of OCR1Dxy, and set the output level on pin TODxyA on compare match A.

TIOR1Dx is initialized to 0000_H by a reset.

(1) OSSDxy[1:0] — Compare Match Output Source Select

OSSDxy		
[1]	[0]	Function
0	0	No TODxyA pin output (Initial value)
0	1	Output level on TODxyA depends on the I/O control bit A on compare match A
1	0	Output level on TODxyA depends on the I/O control bit B on compare match B
1	1	Output level on TODxyA depends on the I/O control bit A or B on compare match A or B, respectively

These bits select the output level on the compare match output pin (TODxyA). The output is controlled by compare match A, compare match B, or both matches.

Either of compare matches A and B is used as a start/stop trigger of timer down counter Dxy (DCNTDxy), and the other one is used as a trigger of compare match output. In addition, both of compare matches A and B are used to control the output, with one as a trigger of an assertion of the output and the other as a trigger of a negation of the output, enabling a one-shot pulse to be output from pin TODxyA.

When both matches are used as a trigger of the output and they occur at the same time, priority is given to compare match B and the output level depends on the IOBxy bit in TIOR2Dx.

A level of 0 is output on TODxyA as a default. While these bits are set to 00_B, the output level on TODxyA is not changed even if compare match A or B occurs.

(2) IOADxy[1:0] — I/O Control A

IOADxy		
[1]	[0]	Function
0	0	Compare match is not performed (Initial value)
0	1	Output level on compare match is 0
1	0	Output level on compare match is 1
1	1	Output level on compare match is toggled

These bits select the function of the output compare register 1Dxy (OCR1Dxy).

When these bits are set to 00_B, compare match between OCR1Dxy and timer counter 1Dx (TCNT1Dx) is not performed. Otherwise, the compare match is performed. An interrupt request is issued on compare match. AD conversion trigger is also issued (only for subblocks D0, D2, D4, D6, and D9).

If compare match A is selected by the compare match output source select bit (OSSDxy), a signal is output on pin TODxyA according to the IOADxy bits (only subblocks D0 to D9).

21.7.2.4 TIOR2Dx — Timer I/O Control Register 2Dx

(x = 0, 1, 2, 3, 4, 5, 6, 7, 8, 9: Corresponding to subblocks D0 to D9, y = 0, 1, 2, 3: Corresponding to channels 0 to 3)

Access: 8-bit and 16-bit accessible

Address: FFE6 1106_H + (100_H * x)

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	IOBDx3			—	IOBDx2			—	IOBDx1			—	IOBDx0		
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Table 21.84 TIOR2Dx Register Contents

Bit Position	Bit Name	Function
15, 11, 7, 3	—	Not used. These bits are always read as 0. When writing, always write 0.
14 to 12, 10 to 8, 6 to 4, 2 to 0	IOBDx3 to IOBDx0	I/O Control B Sets permission/prohibition of the output compare register 2Dxy (OCR2Dxy), and the compare match output value.

The timer I/O Control Register 2Dx (TIOR2Dx) is a 16-bit readable/writable register.

This register has bits to set the functions of the output compare registers 2Dxy(OCR2Dxy). This can enable/disable compare match and can set the output level on pin TODxyA on compare match B.

TIOR2Dx is initialized to 0000_H by a reset.

(1) IOBDxy[2:0] — I/O Control B

IOBDxy			Function		
[2]	[1]	[0]			
0	0	0	Selects compare match output of OCR2Dxy.	Compare match is disabled	(Initial value)
0	0	1		Output level on compare match is 0	
0	1	0		Output level on compare match is 1	
0	1	1		Output level on compare match is toggled	
1	0	0	Reserved	Setting prohibited	
1	0	1		Setting prohibited	
1	1	0		Setting prohibited	
1	1	1		Setting prohibited	

These bits enable and disable of compare match of the output compare register 2Dxy (OCR2Dxy) and select the TODxyA pin output value on the compare match B.

When TIOR2Dx is used as the compare match register (IOBDxy[2] = 0) and the IOBDxy[1:0] bits are set to 00B, compare match between GRDxy and timer counter 2Dx (TCNT2Dx) is not performed. Otherwise, the compare match is performed. At this time, an interrupt request is issued to CPU by compare match. An AD interrupt conversion trigger is also issued (only for subblock D0, D2, D4, D6, D9).

When compare match B is selected as the output source in the compare match output source select bit (OSSDxy), a signal is output on pin TODxyA according to the IOBDxy bits.

21.7.2.5 OSELRDx — Line Select Register Dx

(x = 0, 1, 2, 3, 4, 5, 6, 7, 8, 9: Corresponding to subblocks D0 to D9, y = 0, 1, 2, 3: Corresponding to channels 0 to 3)

Access: 8-bit accessible

Address: FFE6 1111_H + (100_H * x)

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	OSELBDx3	OSELBDx2	OSELBDx1	OSELBDx0	OSELADx3	OSELADx2	OSELADx1	OSELADx0
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.85 OSELRDx Register Contents

Bit Position	Bit Name	Function
7 to 4	OSELBDxy	Output Selection B 0: TODxyB provides normal output. 1: TODxyB provides the value of the output value register.
3 to 0	OSELADxy	Output Selection A 0: TODxyA provides normal output. 1: TODxyA provides the value of the output value register.

The output selection register Dx (OSELRDx) is an 8-bit readable/writable register.

It determines whether to output the value of output value register from the pins (TODxyA and TODxyB).

OSELRDx is initialized to 00_H by a reset.

(1) OSELA (B) Dxy — Output Setting Bits A (B) xy

Writing 1 to this bit outputs the value of the output value register to TODxyA (or TODxyB) pin. Setting of the timer I/O control register 1Dx and 2Dx (TIOR1Dx and TIOR2Dx), setting of the timer output control register Dx (TOCRDx), or one-shot pulse output by the timer downcounter Dxy has no effect on the output value.

When this bit is set to 0, the TODxyA (TODxyB) pin outputs according to the settings of the timer I/O control register 1Dx and 2Dx (TIOR1Dx and TIOR2Dx), and settings of the one-shot pulse output by the timer output control register Dx (TOCRDx).

21.7.2.6 ODRDx — Output Value Register Dx

(x = 0, 1, 2, 3, 4, 5, 6, 7, 8, 9: Corresponding to subblocks D0 to D9, y = 0, 1, 2, 3: Corresponding to channels 0 to 3)

Access: 8-bit accessible

Address: FFE6 1112_H + (100_H * x)

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	ODBDx3	ODBDx2	ODBDx1	ODBDx0	ODADx3	ODADx2	ODADx1	ODADx0
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.86 ODRDx Register Contents

Bit Position	Bit Name	Function
7 to 4	ODBDxy (y = 3, 2, 1, 0: corresponding channel 0 to 3)	Output Value B Sets the value to be output to TODxyB.
3 to 0	ODADxy (y = 3, 2, 1, 0: corresponding channel 0 to 3)	Output Value A Sets the value to be output to TODxyA.

The output value register Dx (ODRDx) is an 8-bit readable/writable register.

When a bit of the output selection register Dx (OSELRDx) is set to 1, the corresponding ODRDx value is issued from respective pins (TODxyA and TODxyB).

ODRDx is initialized to 00_H by a reset.

21.7.2.7 DSTRDx — Down Counter Start Register Dx

(x = 0, 1, 2, 3, 4, 5, 6, 7, 8, 9: Corresponding to subblocks D0 to D9, y = 0, 1, 2, 3: Corresponding to channels 0 to 3)

Access: 8-bit accessible

Address: FFE6 1108_H + (100_H * x)

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	DSTDx3	DSTDx2	DSTDx1	DSTDx0
Value after reset	0	0	0	0	0*1	0*1	0*1	0*1
R/W	R	R	R	R	R/W	R/W	R/W	R/W

Note 1. Value 0 cannot be written. If 1 is written to these bits, data is not retained. This bit is always read as 0.

Table 21.87 DSTRDx Register Contents

Bit Position	Bit Name	Function
7 to 4	—	Not used. These bits are always read as 0. The write value should always be 0. When writing, always write 0.
3 to 0	DSTDxy	Down Counter Start 0: No operation 1: Timer down counters Dxy (DCNTDx3-Dx0) are started

The down counter starting register Dx (DSTRDx) is an 8-bit readable/writable register. This register has a start bit for the down counter. Writing a value to the bit by software starts down counting.

DSTRDx is initialized to 00_H by a reset.

(1) DSTDxy — Down Counter Start Dxy

Setting these bits to 1 makes timer down counter Dxy (DCNTDxy) start. The setting in these bits is always valid and regardless of the start trigger setting in the down counter control register. When compare match B and writing 1 to these bits occurs at the same time if the down counter is set so that it is stopped on compare match B (compare match A or B), compare match B (compare match A or B) takes priority and the down counter is not started.

When DCNTDxy = $0000\ 0000_H$, writing 1 to these bits has no effect.

21.7.2.8 DSR1Dx — Down Counter Status Registers 1Dx

(x = 0, 1, 2, 3, 4, 5, 6, 7, 8, 9: Corresponding to subblocks D0 to D9, y = 0, 1, 2, 3: Corresponding to channels 0 to 3)

Access: 8-bit accessible

Address: FFE6 $110A_H$ + (100_H * x)

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	DWFDx3	DWFDx2	DWFDx1	DWFDx0	DSFDx3	DSFDx2	DSFDx1	DSFDx0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 21.88 DSR1Dx Register Contents

Bit Position	Bit Name	Function
7 to 4	DWFDxy	Down Count Wait Flag Dxy 0: Down count compare match start wait status has not occurred. 1: Down count compare match start wait status has occurred.
3 to 0	DSFDxy	Down Counter Status Flag Dxy 0: Down counter is disabled 1: Down counter is enabled

The down count status register 1Dx (DSR1Dx) is an 8-bit read-only register. This register has a flag that indicates the status of the timer down counter Dxy (DCNTDxy).

DSR1Dx is initialized to 00_H by a reset.

(1) DWFDxy — Down Count Wait Flag Dxy

These bits are flags that indicate whether timer down count Dxy (DCNTDxy) is in the count start wait status. This flag is valid only when compare match A and B is set for the count start trigger by TRGSELDxy bits in the control register Dx (DCRDx). This flag is not set to 1 in other settings.

When compare match A and B is set as a count start trigger and DWFDxy is 0, if compare match A or compare match B occurs, DWFDxy is set to 1. When a compare match (compare match B if DWFDxy is set to 1 on compare match A) of a source different from setting DWFDxy to 1 occurs, it is recognized as a count start trigger and down-counting starts. At this time, the DWFDxy value is cleared to 0.

Even when a compare match of the same source as setting DWFDxy to 1 occurs, the DWFDxy value remains 1 and is not recognized as a count start trigger.

If compare match A and compare match B occur simultaneously, it is immediately recognized as a count start trigger. Therefore, this flag does not enter the down count wait status and DWFDxy is not set to 1.

This flag is read-only flag and cannot be set to 1 or cleared to 0 by the software.

- Setting condition
  - When compare match A or compare match B has occurred while TRGSELDxy in the down counter control register Dx (DCRDx) is 110_B
- Clearing conditions
  - When DWFDxy is set to 1 on compare match A and compare match B has occurred while DWFDxy is 1
  - When DWFDxy is set to 1 on compare match B and compare match A has occurred while DWFDxy is 1
  - Writing 1 to DWFCDxy in the down count status clear register Dx (DSCRDx)

(2) DSFDxy — Down Counter Status Flag Dxy

These bits indicate enabling/disabling of timer down counter Dxy (DCNTDxy). When these bits are read as 1, the counter operation is enabled. If the TDE bit of ATUENR is 1 and the STRDx bit of TSTRD is 1, the counter is running.

When these bits are read as 0, the counter operation is disabled, so counting cannot be in progress.

This flag is a read-only flag; it cannot be set to 1 or cleared to 0 by software.

- Setting conditions
  - When writing 1 to the down count start bit in the down count starting register Dx (DSTRDx)
  - When the condition set as the down counter start trigger is satisfied (compare match A or B)
- Clearing conditions
  - When the down counter is stopped by underflow
  - When the condition to stop the down counter (compare match B)

These flags are set regardless of the settings of the TDE bit in ATUENR and the STRDx bit in TSTRD.

Accordingly, if the TDE bit and the STRDx bit are not set to enable counting, the down counter is not actually running even if these bits indicate that counting is enabled.

Clearing proceeds in synchronization with the down-count clock.

21.7.2.9 DSR2Dx — Down Count Status Register 2Dx

(x = 0, 1, 2, 3, 4, 5, 6, 7, 8, 9: Corresponding to subblocks D0 to D9, y = 0, 1, 2, 3: Corresponding to channels 0 to 3)

Access: 8-bit accessible

Address: FFE6 $110B_H$ + (100_H * x)

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	DWRFDx3[1:0]		DWRFDx2[1:0]		DWRFDx1[1:0]		DWRFDx0[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 21.89 DSR2Dx Register Contents

Bit Position	Bit Name	Function
7 to 0	DWRFDxy[1:0]	Down Count Wait Record Flag Dxy 00: No compare match 01: Down-counting starts on compare match A. 10: Down-counting starts on compare match B. 11: Down-counting starts on simultaneous occurrence of compare match A and compare match B.

Down count status registers 2Dx (DSR2Dx) are 8-bit read-only registers.

The DWRFDxy register is a flag that indicates the condition for starting count operation of the timer down counter Dxy (DCNTDxy).

The status is updated when the timer down counter Dxy (DCNTDxy) starts counting.

When the down count start condition is “A and B”, the down count start (compare match occurring later) condition is retained in the status register. When down-counting is started by the down count start register Dx (DSTRDx), the status register keeps retaining the previous value.

DSR2Dx is initialized to 00_H by a reset.

- Clearing (to 00_B) condition
  Writing 1 to DWRFCDxy in the down count status clear register Dx (DSCRDx)

21.7.2.10 DSCRDx — Down Count Status Clear Register Dx

(x = 0, 1, 2, 3, 4, 5, 6, 7, 8, 9: Corresponding to subblocks D0 to D9, y = 0, 1, 2, 3: Corresponding to channels 0 to 3)

Access: 8-bit accessible

Address: FFE6 1109_H + (100_H * x)

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	DWFCDx3	DWFCDx2	DWFCDx1	DWFCDx0	DWRFCDx3	DWRFCDx2	DWRFCDx1	DWRFCDx0
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.90 DSCRDx Register Contents

Bit Position	Bit Name	Function
7 to 4	DWFCDxy	Down Count Wait Flag Clear Enable Dxy 0: Disabled (Initial value) 1: DWFDxy in the down count status register 1Dx (DSR1Dx) is cleared to 0.
3 to 0	DWRFCDxy	Down Count Wait Record Flag Clear Enable Dxy 0: Disabled (Initial value) 1: DWRFDxy in the down count status register 2Dx (DSR2Dx) is cleared to 0.

The down count status clear register Dx (DSCRDx) is an 8-bit readable/writable register that sets clearing of the flag on compare match when setting compare match A and B for the down count status registers 1Dx, 2Dx (DSR1Dx and DSR2Dx).

DSCRDx is initialized to 00_H by a reset.

(1) DWFCDxy — Down Count Wait Flag Clear Enable Dxy

These bits set clearing the flag that is set by occurrence of down count wait of the down count status register 1Dx (DSR1Dx).

Setting each of these bits to 1 clears the down count wait flag Dxy (DWFDxy) in the down count status register 1Dx (DSR1Dx). These bits are always read as 0.

(2) DWRFCDxy — Down Count Wait Record Flag Clear Enable Dxy

These bits set clearing of the flag that is set by occurrence of down count wait of the down count status register 2Dx (DSR2Dx).

Setting each of these bits to 1 clears the down count wait record flag Dx (DWRFDxy) and the down count wait A flag Dxy (DWRFDxy) in the down count status register 2Dx (DSR2Dx). These bits are always read as 0.

21.7.2.11 DCRDx — Down Counter Control Registers Dx

(x = 0, 1, 2, 3, 4, 5, 6, 7, 8, 9: Corresponding to subblocks D0 to D9, y = 0, 1, 2, 3: Corresponding to channels 0 to 3)

Access: 8-bit and 16-bit accessible

Address: FFE6 1102_H + (100_H * x)

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TRGSELDx3				TRGSELDx2				TRGSELDx1				TRGSELDx0			
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.91 DCRDx Register Contents

Bit Position	Bit Name	Function
15 to 0	TRGSELDx3 to TRGSELDx0	Down Counter Start/Stop Trigger Select Compare match A and compare match B can be set as the start or stop condition for counting of the timer down counter Dxy (DCNTDxy).

Down counter control registers Dx (DCRDx) are 16-bit readable/writable registers that starts the timer down counter Dxy (DCNTDxy). Compare match A, compare match B, compare match A or B, or compare match A and B can be set as a starting or stopping trigger of timer down counters Dxy (DCNTDxy).

DCRDx is initialized to 0000_H by a reset. To change the TRGSELDxy bit, stop the counter. Otherwise operation cannot be guaranteed.

(1) TRGSELDxy — Down Counter Start/Stop Trigger Select Bits Dxy

TRGSELDxy				Function	
[3]	[2]	[1]	[0]	Counter start trigger	Counter stop trigger
0	0	0	0	No trigger	No trigger (Initial value)
0	0	0	1	No trigger	Compare match B
0	0	1	0	Compare match A	No trigger
0	0	1	1	Compare match A	Compare match B
0	1	0	0	Compare match B	No trigger
0	1	0	1	Setting prohibited	Setting prohibited
0	1	1	0	Compare match A and B	No trigger
0	1	1	1	Setting prohibited	Setting prohibited
1	0	0	0	Setting prohibited	Setting prohibited
1	0	0	1	No trigger	Compare match A or B
1	0	1	0	Compare match A or B	No trigger
1	0	1	1	Compare match A or B	Compare match A or B
1	1	0	0	Setting prohibited	Setting prohibited
1	1	0	1	Setting prohibited	Setting prohibited
1	1	1	0	Setting prohibited	Setting prohibited
1	1	1	1	Setting prohibited	Setting prohibited

The trigger sources to start and stop the timer down counter Dxy (DCNTDxy) can be selected by these TRGSELDxy bits. Compare match A, compare match B, compare match A or B, or compare match A and B can be set as a trigger to start counting.

Compare match B or compare match A or B can be set as a trigger to stop counting. In case of compare matches A and B, the counter starts counting down when both compare match A and compare match B are satisfied (compare match A and compare match B need not necessarily occur at the same time).

The counter can also be started by writing 1 to the down counter start bit. The trigger source is always valid regardless of the setting of the TRGSELDxy bits.

If the trigger sources to start and stop the down counter occur at the same time, stopping the counter is given priority.

CAUTION

When an underflow and compare match A occur simultaneously while TRGSELDxy is 1010_B, an underflow interrupt is not generated and the underflow flag is not set to 1. When a count start trigger and a compare match A occur simultaneously when the timer down counter Dxy (DCNTDxy) stops with 0000 0000_H while TRGSELDxy is 1010_B, an underflow interrupt is not generated and the underflow flag is not set to 1.

In this case, to execute software processing for underflow occurrence, write the underflow interrupt flag to 1 (EICz.EIRFz = 1) only when the down counter is stopped (DSR1Dx.DSFDxy = 0) and the underflow interrupt flag of the interrupt controller is 0 (EICz.EIRFz = 0).

21.7.2.12 TSRDx — Timer Status Registers Dx

(x = 0, 1, 2, 3, 4, 5, 6, 7, 8, 9: Corresponding to subblocks D0 to D9, y = 0, 1, 2, 3: Corresponding to channels 0 to 3)

Access: 8-bit and 16-bit accessible

Address: FFE6 $110C_H$ + (100_H * x)

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	OVF2 Dx	OVF1 Dx	UDF Dx3	UDF Dx2	UDF Dx1	UDF Dx0	CMFA Dx3	CMFA Dx2	CMFA Dx1	CMFA Dx0	CMFB Dx3	CMFB Dx2	CMFB Dx1	CMFB Dx0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 21.92 TSRDx Register Contents

Bit Position	Bit Name	Function
15, 14	—	Not used. These bits are always read as 0. When writing, always write 0.
13	OVF2Dx	Overflow Flag 2Dx 0: TCNT2Dx has not overflowed 1: TCNT2Dx has overflowed
12	OVF1Dx	Overflow Flag 1Dx 0: TCNT1Dx has not overflowed 1: TCNT1Dx has overflowed
11 to 8	UDFDx3 to UDFDx0	Underflow Flag Dx3-Dx0 0: DCNTDxy has not underflowed 1: DCNTDxy has underflowed
7 to 4	CMFADx3 to CMFADx0	Compare Match A Flag Dx3-Dx0 0: Compare match A has not occurred 1: Compare match A has occurred
3 to 0	CMFBDx2 to CMFBDx0	Compare Match B Flag Dx3-Dx0 0: Compare match B has not occurred 1: Compare match B has occurred

The timer status register Dx (TSRDx) is a 16-bit read-only register. This register indicates overflow of two timer counters 2Dx, 1Dx (TCNT2Dx and TCNT1Dx) of subblock Dx (x = 0, 1, 2, 3, 4, 5, 6, 7, 8, 9), underflow of the timer down counters Dxy (DCNTDxy) of channels 0 to 3, and compare match of the output compare registers 1Dxy, 2Dxy (OCRD1xy and OCR2Dxy). If an interrupt source takes effect when this flag is set, an interrupt request is made again. An interrupt request is made even if there is a conflict between clearing by the corresponding timer status clear register and setting by an interrupt source.

Overflow flags 2 and 1 as well as underflow flags and compare match flags A and B can be cleared by setting the corresponding bits of the timer status clear register Dx (TSCRDx). Each of compare match A and B can use 20 pulse outputs (subblocks D0, D2, D4, D6, and D9) as the AD conversion trigger. TSRDx is initialized to 0000_H by a reset.

(1) OVF2Dx — Overflow Flag 2Dx

OVF2Dx	Function
0	[Clearing condition] When 1 is written to the OVFC2Dx bit of the timer status clear register Dx (TSCRDx). (Initial value)
1	[Setting condition] • When TCNT2Dx overflowed (from FFFF FFFF$_H$ to 0000 0000$_H$) • When the counter is cleared with the upper-limit setting function enabled

Indicates whether or not timer counter 2Dx (TCNT2Dx) has overflowed. If the reading value of this flag is 1, it means that an overflow in TCNT2Dx has occurred. This flag cannot be set to 1 by software. The overflow flag is set when the count-up clock is input while TCNT2Dx is FFFF FFFF$_H$ or at the first clock after the counter value matches the upper-limit setting register value with the upper-limit setting function enabled. Writing 0000 0000$_H$ to TCNT2Dx or starting TCNT2Dx from an initial value of 0000 0000$_H$ has no effect on this bit.

When writing to TCNT2Dx at the same time as incrementation while it is FFFF FFFF$_H$, this bit is set to 1. However, TCNT2Dx is updated with the written value, but not 0000 0000$_H$.

When an assertion of counter clearing signal from timer B and overflow due to count up occur, overflow does not occur.

Simultaneous occurrence of counter clearing by timer B and counter clearing with the upper-limit setting function enabled is detected as an overflow.

(2) OVF1Dx — Overflow Flag 1Dx

OVF1Dx	Function
0	[Clearing condition] When 1 is written to the OVFC1Dx bit of the timer status clear register Dx (TSCRDx). (Initial value)
1	[Setting condition] • When TCNT1Dx overflowed (from FFFF FFFF$_H$ to 0000 0000$_H$) • When the counter is cleared with the upper-limit setting function enabled

Indicates whether or not timer counter 1Dx (TCNT1Dx) has overflowed. If the reading value of this flag is 1, it means that an overflow in TCNT1Dx has occurred. This flag cannot be set to 1 by software. The overflow flag is set when the count-up clock is input while TCNT1Dx is FFFF FFFF$_H$ or at the first clock after the counter value matches the upper-limit setting register value with the upper-limit setting function enabled. Writing 0000 0000$_H$ to TCNT1Dx or starting TCNT1Dx from an initial value of 0000 0000$_H$ has no effect on this bit.

When writing to TCNT1Dn at the same time as incrementation while it is FFFF FFFF$_H$, this bit is set to 1.

However, TCNT1Dx is updated with the written value, but not 0000 0000$_H$. When an assertion of counter clearing signal from timer B and an overflow by counting up occur simultaneously, the overflow is not detected. Simultaneous occurrence of counter clearing by timer B and counter clearing with the upper-limit setting function enabled is detected as an overflow

(3) UDFDxy — Underflow Flag Dxy

UDFDxy	Function
0	[Clearing condition] When 1 is written to the UDFCDxy bit of the timer status clear register Dx (TSCRDx). (Initial value)
1	[Setting condition] When DCNTDxy underflowed (decremented while DCNTDxy = 0000 0000_H)

This flag is a status flag that indicates underflow of subblock Dx and down counter Dxy of channel y. If the reading value of this flag is 1, it means that DCNTDxy had an underflow. These flags cannot be set to 1 by software.

This bit is set to 1 when DCNTDxy is to be decremented while it is 0000 0000_H. DCNTDxy holds 0000 0000_H on underflow. Writing FFFF $FFFF_H$ to DCNTDxy has no effect on these bits.

These bits are initialized to 0 by a reset. Although DCNTDxy is initialized to 0000 0000_H, these flags do not indicate underflow because DCNTDxy has not started. For details on control of DCNTDxy, see descriptions of timer D down counters.

(4) CMFADxy — Compare Match A Flag Dxy

CMFADxy	Function
0	[Clearing condition] When 1 is written to the CMFCADxy bit of the timer status clear register Dx (TSCRDx). (Initial value)
1	[Setting condition] When the values in timer counter 1Dx (TCNT1Dx) and the output compare register 1Dxy (OCR1Dxy) match while operation of compare match with OCR1Dxy is enabled

These bits indicate whether or not compare match between the output compare register 1Dxy (OCR1Dxy) and TCNT1Dx has occurred. If the reading value of this flag is 1, it means that a compare match occurs between TCNT1Dx and OCR1Dxy. This flag cannot be set to 1 by software.

When operation of compare match between OCRDxy and TCNT1Dx is enabled by the setting in timer I/O control register 1Dx (TIOR1Dx), compare match operation is performed regardless of the state of TCNT1Dx. These bits are set to 1 on the first edge of the PCLK after the values in TCNT1Dx and OCR1Dxy match. Even if these compare match flags are cleared to 0 by software while TCNT1Dx = OCR1Dxy after the match in comparison is detected, these bits are not set to 1 again.

A single pulse, signaling detection of compare match A, whose width is equal to the cycle of the PCLK is output to activate the A/D converter (supported by 20 channels in subblocks D0, D2, D4, D6, D9).

If TCNT1Dx matches OCR1Dxy again before the status flag is cleared, the compare match A is detected and the status flag is rewritten with 1.

(5) CMFBDxy — Compare Match B Flag Dxy

CMFBDxy	Function
0	[Clearing condition] When 0 is written to the CMFCBDxy bit of the timer status clear register Dx (TSCRDx). (Initial value)
1	[Setting condition] When the values in timer counter 2Dx (TCNT2Dx) and output compare register 2Dxy (OCR2Dxy) match while operation of compare match with OCR2Dxy is enabled

These bits indicate whether or not compare match between the output compare register 2Dxy (OCR2Dxy) and TCNT2Dx has occurred. If the reading value of this flag is 1, it means that a compare match of TCNT2Dx and OCR2Dxy has happened. This flag cannot be set to 1 by software.

When operation of compare match between OCR2Dxy and TCNT2Dx is enabled by the setting in timer I/O control register 2Dx (TIOR2Dx), compare match operation is performed regardless of the state of TCNT2Dx. These bits are set to 1 on the first edge of the Pϕ clock after the values in TCNT2Dx and OCR2Dxy match.

Even if these compare match flags are cleared to 0 by software while TCNT2Dx = OCR2Dxy after the match in comparison is detected, these bits are not set to 1 again.

A single pulse, signaling detection of compare match B, whose width is equal to the cycle of the PCLK is output to activate the A/D converter (supported by 20channels in subblocks D0, D2, D4, D6, D9).

If TCNT2Dx matches OCR2Dxy again before the status flag is cleared, the compare match B is detected and the status flag is rewritten with 1.

21.7.2.13 TSCRDx — Timer Status Clear Register Dx

(x = 0, 1, 2, 3, 4, 5, 6, 7, 8, 9: Corresponding to subblocks D0 to D9, y = 0, 1, 2, 3: Corresponding to channels 0 to 3)

Access: 8-bit access and 16-bit access are possible.
Value 0 cannot be written. The value is not retained when 1 is written. This bit is always read as 0.

Address: FFE6 $110E_H$ + (100_H * x)

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	OVFC2Dx	OVFC1Dx	UDFCDx3	UDFCDx2	UDFCDx1	UDFCDx0	CMFCADx3	CMFCADx2	CMFCADx1	CMFCADx0	CMFCBDx3	CMFCBDx2	CMFCBDx1	CMFCBDx0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.93 TSCRDx Register Contents

Bit Position	Bit Name	Function
15, 14	—	Not used. These bits are always read as 0. When writing, always write 0.
13	OVFC2Dx	Overflow Flag Clear Enable 2Dx 0: Disabled (Initial value) 1: Clears OVF2Dx of timer status register Dx (TSRDx) to 0.
12	OVFC1Dx	Overflow Flag Clear Enable 1Dx 0: Disabled (Initial value) 1: Clears OVF1Dx of timer status register Dx (TSRDx) to 0.
11 to 8	UDFCDx3 to UDFCDx0	Underflow Flag Clear Enable Dx3-Dx0 0: Disabled (Initial value) 1: Clears UDFDx3-UDFDx0 of timer status register Dx (TSRDx) to 0.
7 to 4	CMFCADx3 to CMFCADx0	Compare Match A Flag Clear Enable Dx3-Dx0 0: Disabled (Initial value) 1: Clears CMFADx3-CMFADx0 of timer status register Dx (TSRDx) to 0.
3 to 0	CMFCBDx3 to CMFCBDx0	Compare Match B Flag Clear Enable Dx3-Dx0 0: Disabled (Initial value) 1: Clears CMFBDx3-CMFBDx0 of timer status register Dx (TSRDx) to 0.

The timer status clear register Dx (TSCRDx) is a 16-bit readable/writable register. This register specifies the flag clearing at the time of overflow of two 32-bit up-counters (TCNT2Dx and TCNT1Dx), compare match of TCNT1Dx and the output compare register 1Dxy (OCR1Dxy), and compare match of TCNT2Dx and the output compare register 2Dxy (OCR2Dxy).

When TSCRDx is reset, it is initialized to 0000_H.

(1) OVFC2Dx — Overflow Flag Clear Enable 2Dx

This bit specifies flag clearing at the time of overflow of the timer counter 2Dx (TCNT2Dx).

Setting this bit enables clearing of the overflow flag 2Dx (OVF2Dx) of the timer status register Dx (TSRDx). This bit is always read as 0.

OVFC2Dx	Function	
0	Disabled	(Initial value)
1	Clears OVF2Dx to 0.	

(2) OVFC1Dx — Overflow Flag Clear Enable 1Dx

This bit specifies flag clearing at the time of overflow of the timer counter 1Dx (TCNT1Dx).

Setting this bit enables clearing of overflow flag 1Dx (OVF1Dx) of the timer status register Dx (TSRDx). This bit is always read as 0.

OVFC1Dx	Function	
0	Disabled	(Initial value)
1	Clears OVF1Dx to 0.	

(3) UDFCDxy — Underflow Flag Clear Enable Dxy

This bit specifies flag clearing at the time of underflow of the timer down counter Dxy (DCNTDxy).

Setting this bit enables clearing of the underflow flag Dxy (UDFDxy) of the timer status register Dx (TSRDx). This bit is always read as 0.

UDFCDxy	Function	
0	Disabled	(Initial value)
1	Clears UDFDxy to 0.	

(4) CMFCADxy — Compare Match A Flag Clear Enable Dxy

This flag specifies flag clearing at the time of compare match of the output compare register 1Dxy (OCR1Dxy). Setting this bit enables clearing of the compare match A flag Dxy (CMFADxy) of the timer status register Dx (TSRDx). This bit is always read as 0.

CMFCADxy	Function	
0	Disabled	(Initial value)
1	Clears CMFADxy to 0.	

(5) CMFCBDxy — Compare Match B Flag Clear Enable Dxy

This flag specifies flag clearing at the time of compare match of the output compare register 2Dxy (OCR2Dxy). Setting this bit enables clearing of the compare match B flag Dxy (CMFBDxy) of the timer status register Dx (TSRDx). This bit is always read as 0.

CMFCBDxy	Function	
0	Disabled	(Initial value)
1	Clears CMFBDxy to 0.	

21.7.2.14 TOCRDx — Timer Output Control Registers Dx

(x = 0, 1, 2, 3, 4, 5, 6, 7, 8, 9: Corresponding to subblocks D0 to D9, y = 0, 1, 2, 3: Corresponding to channels 0 to 3)

Access: 8-bit access is possible.

Address: FFE6 1110_H + (100_H * x)

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TONEBDx	TONEADx
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 21.94 TOCRDx Register Contents

Bit Position	Bit Name	Function
7 to 2	—	Not used. These bits are always read as 0. When writing, always write 0.
1	TONEBDx	Output Inversion Select TODxyB 0: Output level is not inverted 1: Output level is inverted
0	TONEADx	Output Inversion Select TODxyA 0: Output level is not inverted 1: Output level is inverted

The timer output control register Dx (TOCRD0 to TOCRD9) is an 8-bit readable/writable register. This register specifies whether to invert signals of the output pins TODxyA and TODxyB of each subblock Dx.

When this register is reset, it is initialized to 00_H.

(1) TONEBDx, TONEADx — Output Inversion Select Dx

Specifies whether to invert the output from TODxy and TODxyA.

TONEBDx	Function	
0	Normal output from the output pin (TODxyB)	(Initial value)
1	Inverted output from the output pin (TODxyB)	

TONEADx	Function	
0	Normal output from the output pin (TODxyA)	(Initial value)
1	Inverted output from the output pin (TODxyA)	

Signals on pins TODxyA and TODxyB are inverted on the first edge of the PCLK after the output inversion select Dx bit is set to 1.

This function is not affected by the operating state of timer counters 1Dx and 2Dx (TCNT1Dx, TCNT2Dx).

The TONEBDx bit controls four outputs (TODx0B, TODx1B, TODx2B, TODx3B) in a single subblock. The TONEADx bit controls four outputs (TODx0A, TODx1A, TODx2A, TODx3A) in a single subblock. These bits cannot control individual signals independently (channel control is not available).

Output levels on pins TODxyA and TODxyB are initialized to a level of 0 (when TONEADx = 0 and TONEBDx = 0).

21.7.2.15 OSBRDx — Timer Offset Base Registers Dx

(x = 0, 1, 2, 3, 4, 5, 6, 7, 8, 9: Corresponding to subblocks D0 to D9)

Access: 32-bit read access is supported. 8-bit access and 16-bit access are prohibited.

Address: FFE6 1118$_H$ + (100$_H$ * x)

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	OSBRD[31:16]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PSBRD[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Note 1. This register does not accept write access.

Timer offset base registers Dx (OSBRDx) are 32-bit read only registers that are used only for input capture. The timer counter 1Dx (TCNT1Dx) value is captured and stored by a trigger signal from timer A and PH update notification triggers 0 and 1 from DFE.

The trigger signal from timer A is selectable from TIA00 to TIA06, and PH update notification triggers 0 and 1 from DFE as external input signals are selectable in the timer input capture trigger select register Dx (TICTSELDx). For details, see **Section 21.4.2.1, TCR1A — Timer Control Register 1A** and **Section 21.7.2.16, TICTSELDx — Timer Input Capture Trigger Select Register Dx**.

When this register is reset, it is initialized to 0000 0000$_H$.

21.7.2.16 TICTSELDx — Timer Input Capture Trigger Select Register Dx

(x = 0, 1, 2, 3, 4, 5, 6, 7, 8, 9: Corresponding to subblocks D0 to D9)

Access: 8-bit accessible

Address: FFE6 1114_H + (100_H * x)

Value after reset: $0X_H$

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TIDSELDx[1:0]	
Value after reset	0	0	0	0	0	0	0	*1
R/W	R	R	R	R	R	R	R/W	R/W

Note 1. This bit varies with the subblocks.
x = 0 to 2: 0
x = 3 to 9: 1

The timer input capture trigger select register Dx (TICTSELDx) is an 8-bit readable/writable register.

This register is used to select event 2A or 2B from timer A and PH update notification triggers 0 and 1 from DFE as external input signals, and to capture the TCNT1Dx counter value in the OSBRDx register.

Table 21.95 TICTSELDx Register Contents

Bit Position	Bit Name	Function
7 to 2	—	Not used. These bits are read as 0. When writing, always write 0.
1, 0	TIDSELDx[1:0]	00: Event 2A 01: Event 2B 10: PH update notification trigger 0 from the DFE 11: PH update notification trigger 1 from the DFE

21.7.2.17 TCNT1Dx — Timer Counter 1Dx

(x = 0, 1, 2, 3, 4, 5, 6, 7, 8, 9: Corresponding to subblocks D0 to D9)

Access: 32-bit access is supported. 8-bit access and 16-bit access are prohibited.

Address: FFE6 1120_H + (100_H * x)

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TCNT1D[31:16]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TCNT1D[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Timer counters 1Dx (TCNT1Dx) are 32-bit readable/writable registers driven by the clock selected in the CKSEL1Dx[2:0] bits in timer control register Dx (TCRDx). These counters are started by setting the bit in timer start register D (TSTRD) to 1.

An overflow interrupt request can be issued to CPU when the timer overflows.

The overflow flag 1Dx (OVF1Dx) of the timer status register Dx (TSRDx) is set to 1. When this register is reset, it is initialized to 0000 0000_H.

21.7.2.18 TCNT2Dx — Timer Counter 2Dx

(x = 0, 1, 2, 3, 4, 5, 6, 7, 8, 9: Corresponding to subblocks D0 to D9)

Access: 32-bit access is supported. 8-bit access and 16-bit access are prohibited.

Address: FFE6 1124_H + (100_H * x)

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TCNT2D[31:16}															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TCNT2D[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Timer counters 2Dx (TCNT2Dx) are 32-bit readable/writable registers driven by the clock selected in the CKSEL2Dx[2:0] bits in timer control register Dx (TCRDx). These counters are started by setting the bit in timer start register D (TSTRD) to 1.

An overflow interrupt request can be issued to CPU when the timer overflows.

The overflow flag 2Dx (OVF2Dx) of the timer status register Dx (TSRDx) is set to 1. When this register is reset, it is initialized to 0000 0000_H.

21.7.2.19 CUCR1Dx — Counter Upper-Limit Setting Compare Register 1Dx

(x = 0, 1, 2, 3, 4, 5, 6, 7, 8, 9: Corresponding to subblocks D0 to D9)

Access: 32-bit accessible but 8-bit/16-bit inaccessible

Address: FFE6 1128_H + (100_H * x)

Value after reset: FFFF $FFFF_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CUCR1D[31:16]															
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CUCR1D[15:0]															
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

The counter upper-limit setting compare register 1Dx (CUCR1Dx) is a 32-bit readable/writable register that has a function for comparison with the TCNT1Dx counter. Compare match between this register and the TCNT1Dx counter is enabled by setting the TCNT1Dx clear setting bit (CLR1Dx) in the timer control register Dx (TCRDx) to 1. The TCNT1Dx counter value is cleared to 0000 0000_H on compare match between timer counter 1Dx (TCNT1Dx) and counter upper-limit setting compare register 1Dx (CUCR1Dx). When this compare match and count up of the timer counter 1Dx (TCNT1Dx) occur simultaneously, the timer counter 1Dx (TCNT1Dx) is cleared to 0000 0001_H.

The OVF1Dx bit in timer status register Dx (TSRDx) is set to 1 and an overflow interrupt is output on compare match between timer counter 1Dx (TCNT1Dx) and counter upper-limit setting compare register 1Dx (CUCR1Dx).

The overflow interrupt output width is one clock width of PCLK.

OCR1Dxy compare match by the counter upper-limit setting compare register 1Dx (CUCR1Dx) occurs when CUCR1Dx ≥ OCR1Dxy.

Do not set CUCR1Dx to 0000 0000_H. Note that if CUCR1Dx is set to 0000 0000_H, a compare match occurs at incorrect intervals.

21.7.2.20 CUCR2Dx — Counter Upper-Limit Setting Compare Register 2Dx

(x = 0, 1, 2, 3, 4, 5, 6, 7, 8, 9: Corresponding to subblocks D0 to D9)

Access: 32-bit accessible but 8-bit/16-bit inaccessible

Address: FFE6 $112C_H$ + (100_H * x)

Value after reset: $FFFF\ FFFF_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CUCR2D[31:16]															
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CUCR2D[15:0]															
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

The counter upper-limit setting compare register 2Dx (CUCR2Dx) is a 32-bit readable/writable register that has a function for comparison with the TCNT2Dx timer counter. Compare match between this register and the TCNT2Dx counter is enabled by setting the TCNT2Dx clear setting bit (CLR2Dx) in the timer control register Dx (TCRDx) to 1. The TCNT2Dx counter value is cleared to $0000\ 0000_H$ on compare match between the timer counter 2Dx (TCNT2Dx) and counter upper-limit setting compare register 2Dx (CUCR2Dx).When this compare match and count up of the timer counter 2Dx (TCNT2Dx) occur simultaneously, the timer counter 2Dx (TCNT2Dx) is cleared to $0000\ 0001_H$.

The OVF2Dx bit in timer status register Dx (TSRDx) is set to 1 and an overflow interrupt is output on compare match between the timer counter 2Dx (TCNT2Dx) and counter upper-limit setting compare register 2Dx (CUCR2Dx).

The overflow interrupt output width is one clock width of PCLK.

OCR2Dxy compare match by the counter upper-limit setting compare register 2Dx (CUCR2Dx) occurs when CUCR2Dx ≥ OCR2Dxy.

Do not set CUCR2Dx to $0000\ 0000_H$. Note that if CUCR2Dx is set to $0000\ 0000_H$, a compare match occurs at incorrect intervals.

21.7.2.21 OCR1Dxy — Output Compare Registers 1Dxy

(x = 0, 1, 2, 3, 4, 5, 6, 7, 8, 9: Corresponding to subblocks D0 to D9, y = 0, 1, 2, 3: Corresponding to channels 0 to 3)

Access: 32-bit access is supported. 8-bit access and 16-bit access are prohibited.

Address: FFE6 1140_H + (100_H * x) + (20_H * y)

Value after reset: FFFF $FFFF_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	OC1D[31:16]															
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OC1D[15:0]															
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

OCR1Dxy are 32-bit readable/writable registers.

OCR1Dxy is constantly compared with the timer counter 1Dx (TCNT1Dx). When compare match operation is selected by bit IOADxy in TIOR1Dx, the compare match A flag (CMFADxy) in TSRDx is set to 1 on the first edge of the PCLK after the values in TCNT1Dx and OCR1Dxy match. When compare match A is selected by an output source select bit, a signal is output to pin TODxyA on compare match.

When compare match A is selected as a down counter starting trigger by the TRGSELDxy bit in DCRDxy, DCNTDxy is ready to be counted down on compare match A.

When the down counter (DCNTDxy) is ready, it is started in synchronization with the down counter clock. At this time, a one-shot pulse can be output on pin TODxyB. If compare match A and down-counter stop trigger are output at the same time, output is disabled without any pulse.

When TCNT1Dx overflows to change from FFFF $FFFF_H$ to 0000 0000_H and OCRDxy is set to 0000 0000_H, the match in comparison is detected.

If the counter upper-limit setting function (see **Section 21.7.3.2, Counter Upper-Limit Setting Function**) is in use with clock-bus line 5 or 6, or with a division ratio of 1/1 selected for the prescaler, the timer counter counts between 1 and the value of CUCR1Dx and the counter value may not become 0000 0000_H. Therefore, do not set these registers to 0000 0000_H in such cases.

An interrupt can be issued to CPU by a compare match A detection. Subblocks 0, 2, 4, 6, and 9 can issue AD start triggers.

When the range compare function is enabled (RCR1Dx is not 0), the TCNT1Dx value at the compare match timing is stored in OCR1Dxy.

This register is initialized to FFFF $FFFF_H$ by a reset.

21.7.2.22 RCR1Dx — Range Comparison Value Setting Register 1Dx

(x = 0, 1, 2, 3, 4, 5, 6, 7, 8, 9: Corresponding to subblocks D0 to D9)

Access: 8-bit accessible

Address: FFE6 1130_H + (100_H * x)

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	RC1D[4:0]				
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

The range comparison value setting register 1Dx (RCR1Dx) is an 8-bit readable/writable register.

According to the setting of this register, if the TCNT1Dx value is within the range of OCR1Dxy ≤ TCNT1Dx ≤ (OCR1Dxy + the value of the range specified by RCR1Dx) when writing to OCR1Dxy, a compare match A between OCR1Dxy and TCNT1Dx is generated. RCR1Dx is initialized to 00_H by a reset.

This function is enabled when RCR1Dx is set to a value other than 00_H, and is judged only when a value is written to OCR1Dxy.

When setting the TCNT1Dx clear 1Dx bit (CLR1Dx) to 1, set RCR1Dx to 00_H.

The following table shows ranges that can be set as range compare values (power of 2 (excluded only when RCR1Dx = 0)).

RCR1Dx[4:0]	Selectable Range	Note
00000	None	Range compare disable (Initial value)
00001	2^1-1	—
00010	2^2-1	—
00011	2^3-1	—
00100	2^4-1	—
00101	2^5-1	—
00110	2^6-1	—
00111	2^7-1	—
01000	2^8-1	—
01001	2^9-1	—
01010	$2^{10}-1$	—
01011	$2^{11}-1$	—
01100	$2^{12}-1$	—
01101	$2^{13}-1$	—
01110	$2^{14}-1$	—
01111	$2^{15}-1$	—
10000	$2^{16}-1$	—
10001	$2^{17}-1$	—
10010	$2^{18}-1$	—
10011	$2^{19}-1$	—
10100	$2^{20}-1$	—
10101	$2^{21}-1$	—
10110	$2^{22}-1$	—
10111	$2^{23}-1$	—
11000	$2^{24}-1$	—
11001	$2^{25}-1$	—
11010	$2^{26}-1$	—
11011	$2^{27}-1$	—
11100	$2^{28}-1$	—
11101	$2^{29}-1$	—
11110	$2^{30}-1$	—
11111	$2^{31}-1$	—

21.7.2.23 OCR2Dxy — Output Compare Registers 2Dxy

(x = 0, 1, 2, 3, 4, 5, 6, 7, 8, 9: Corresponding to subblocks D0 to D9, y = 0, 1, 2, 3: Corresponding to channels 0 to 3)

Access: 32-bit access is supported. 8-bit access and 16-bit access are prohibited.

Address: FFE6 1148_H + (100_H * x) + (20_H * y)

Value after reset: FFFF $FFFF_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	OC2D[31:16]															
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OC2D[15:0]															
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

OCR2Dxy are 32-bit readable/writable registers.

OCR2Dxy is always compared with timer counter T2Dx (CNT2Dx), and if the IOBDxy bit of the TIOR2Dx register permits compare match, a match of TCNT2Dx and OCR2Dxy values is detected. If the OSSDxy bit specifies compare match B as the output source, the compare match is output to the TODxyA pin at the next PCLK cycle.

When compare match B is selected as a down counter starting trigger by the TRGSELDxy bit in DCRDx, DCNTDxy is ready to be counted down on compare match B.

When the down counter (DCNTDxy) is ready, it is started in synchronization with the down counter clock. At this time, a one-shot pulse can be output on pin TODxyB. If compare match A and down-counter stop trigger are output at the same time, output is disabled without any pulse.

If the TRGSELDxy bit of DCRDx specifies compare match B as the down count stop trigger, detection of compare match B disables down counting. At the next down-counting clock cycle, the down counter is cleared to 0 and TODxyB output (one-shot pulse) is negated.

When TCNT2Dx overflows to change from FFFF $FFFF_H$ to 0000 0000_H and OCR2Dxy is set to 0000 0000_H, the match in comparison is detected.

If the counter upper-limit setting function (see **Section 21.7.3.2, Counter Upper-Limit Setting Function**) is in use with clock-bus line 5 or 6, or with a division ratio of 1/1 selected for the prescaler, the timer counter counts between 1 and the value of CUCR2Dx and the counter value may not become 0000 0000_H. Therefore, do not set these registers to 0000 0000_H in such cases.

An interrupt can be issued to CPU by compare match B detection. Subblocks 0, 2, 4, 6, and 9 can issue AD start triggers.

When the range compare function is enabled (RCR2Dx is not 0), the TCNT2Dx value at the compare match timing is stored in OCR2Dxy.

This register is initialized to FFFF $FFFF_H$ by a reset.

21.7.2.24 RCR2Dx — Range Comparison Value Setting Register 2Dx

(x = 0, 1, 2, 3, 4, 5, 6, 7, 8, 9: Corresponding to subblocks D0 to D9)

Access: 8-bit accessible

Address: FFE6 1134_H + (100_H * x)

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	RC2D[4:0]				
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

The range comparison value setting register 2Dx (RCR2Dx) is a 5-bit readable/writable register.

According to the setting of this register, if the TCNT2Dx value is within the range of OCR2Dxy ≤ TCNT2Dx ≤ (OCR2Dxy + the value of the range specified by RCR2Dx) when writing to OCR2Dxy, a compare match B between OCR2Dxy and TCNT2Dx is generated. RCR2Dx is initialized to 00_H by a reset.

This function is enabled when RCR2Dx is set to a value other than 00_H, and is judged only when a value is written to OCR2Dxy.

When setting the TCNT2Dx clear 2Dx bit (CLR2Dx) to 1, set RCR2Dx to 00_H.

The following table shows ranges that can be set as range compare values (power of 2 (excluded only when RCR2Dx = 0)).

RCR2Dx[4:0]	Selectable Range	Note
00000	None	Range compare is invalid (Initial value)
00001	2^1-1	—
00010	2^2-1	—
00011	2^3-1	—
00100	2^4-1	—
00101	2^5-1	—
00110	2^6-1	—
00111	2^7-1	—
01000	2^8-1	—
01001	2^9-1	—
01010	$2^{10}-1$	—
01011	$2^{11}-1$	—
01100	$2^{12}-1$	—
01101	$2^{13}-1$	—
01110	$2^{14}-1$	—
01111	$2^{15}-1$	—
10000	$2^{16}-1$	—
10001	$2^{17}-1$	—
10010	$2^{18}-1$	—
10011	$2^{19}-1$	—
10100	$2^{20}-1$	—
10101	$2^{21}-1$	—
10110	$2^{22}-1$	—
10111	$2^{23}-1$	—
11000	$2^{24}-1$	—
11001	$2^{25}-1$	—
11010	$2^{26}-1$	—
11011	$2^{27}-1$	—
11100	$2^{28}-1$	—
11101	$2^{29}-1$	—
11110	$2^{30}-1$	—
11111	$2^{31}-1$	—

21.7.2.25 ICR1Dxy — Input capture register 1Dxy

(x = 0, 1, 2, 3, 4, 5, 6, 7, 8, 9: Corresponding to subblocks D0 to D9, y = 0, 1, 2, 3: Corresponding to channels 0 to 3)

Access: 32-bit access is supported. 8-bit access and 16-bit access are prohibited.

Address: FFE6 1150_H + (100_H * x) + (20_H * y)

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	IC1D[31:16]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IC1D[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

The input capture register 1Dxy (ICR1Dxy) is a 32-bit read-only register, and its initial value is 0000 0000_H. This register captures the counter value of TCNT1Dx into ICR1Dxy triggered by compare match B.

When ICR1Dxy is reset, it is initialized to 0000 0000_H.

21.7.2.26 ICR2Dxy — Input capture register 2Dxy

(x = 0, 1, 2, 3, 4, 5, 6, 7, 8, 9: Corresponding to subblocks D0 to D9, y = 0, 1, 2, 3: Corresponding to channels 0 to 3)

Access: 32-bit access is supported. 8-bit access and 16-bit access are prohibited.

Address: FFE6 1154_H + (100_H * x) + (20_H * y)

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	IC2D[31:16]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	IC2D[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

The input capture register 2Dxy (ICR2Dxy) is a 32-bit read-only register, and its initial value is 0000 0000_H. This register captures the counter value of TCNT2Dx in ICR2Dxy triggered by compare match A.

When ICR2Dxy is reset, it is initialized to 0000 0000_H.

21.7.2.27 DCNTDxy — Timer Down Counters Dxy

(x = 0, 1, 2, 3, 4, 5, 6, 7, 8, 9: Corresponding to subblocks D0 to D9, y = 0, 1, 2, 3: Corresponding to channels 0 to 3)

Access: 32-bit access is supported. 8-bit access and 16-bit access are prohibited.

Address: FFE6 1158_H + (100_H * x) + (20_H * y)

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CNTD[31:16]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CNTD[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Timer down counters Dxy (DCNTDxy) are 32-bit readable/writable registers driven by the clock selected in the DCSELDx[2:0] bits in timer control register Dx (TCRDx).

DCNTDxy is controlled by the down counter control register Dx (DCRDx). Down-counting is started when compare match A, compare match B, compare match A or B, or compare match A and B is detected or when the DSTDxy bit in the DSTRDx register is set to 1. While the down counter is enabled, it is decremented every input of the down counter clock. Counting can be stopped by an underflow of DCNTDxy, detection of compare match B, or detection of compare match A or B. When the decrementing is stopped by an underflow, a down counter underflow interrupt request and a DMA transfer request can be issued.

Decrementation is enabled; on the first edge of the PCLK (the same as compare match A) after the values in TCNT1Dx and OCRDxy match; on the first edge of the PCLK (the same as compare match B) after the values in TCNT2Dx and OCR2Dxy match; on the first edge of the PCLK after the DSTDxy bit is set to 1. Decrementation is enabled until DCNTDxy underflows or until the first edge of the PCLK (the same as compare match A) after the values in TCNT1Dx and OCR2Dxy or the first edge of the PCLK (the same as compare match B) after the values in TCNT2Dx and OCR2Dxy match. The down counter is decremented every input of the down counter clock while it is enabled.

Once DCNTDxy is enabled, it remains enabled until DCNTDxy underflow, compare match B, or compare match A or B (if selected as a count stop trigger) is detected. While it is enabled, another counter starting trigger or writing 1 to DSTDxy bit has no effect on the enabled state of the counter.

When compare match B or compare match A or B (if selected as a count stop trigger) is detected, the counter is stopped on the first edge of the down counter clock and then cleared to 0000 0000_H. If a down count start trigger (compare match A or writing 1 to DSTDxy) and a down count stop trigger (compare match B) are detected at the same time, the count stop trigger takes priority. The counter is not decremented and no signal on pin TODxyB is output. Moreover, When no down counter clock is input during the time between enabled state due to compare match A or writing 1 to DSTDxy and counter stop state due to compare match B, the counter stops without performing down-count. The counter stops down-counting in the down-count stop state regardless of that the DCNTDxy value is 0000 0000_H.

If a value other than 0000 0000_H is set in DCNTDxy after decrementation is terminated by underflow, the counter is not decremented until the counter stopping source is activated. When the DCNTDxy value is 0000 0000_H and the down-count clock is not input at the beginning of down-counting, only flags in the down count status register 2Dx (DSR2Dx) are updated without performing down-counting. A count down stop interrupt (due to DCNTDxy = 0000 0000_H) is generated at the next down-count clock. The down-count status flag Dxy (DSFDxy) in the down count status register 1Dx (DSR1Dx) retains 1 until an interrupt occurs.

This is initialized to 0000 0000_H by a reset.

21.7.3 Operation

Timer D consists of 10 subblocks. Each block has two range compare setting registers (RCR1Dx and RCR2Dx), two 32-bit timer counters 1Dx, 2Dx (TCNT1Dx and TCNT2Dx), one timer offset base register Dx (OSBRDx), one output value register (ODRDx), one output select register (OSELRDx), and four channels described below.

Each channel consists of the output compare register 1Dxy (OCR1Dxy) to which TCNT1Dx is compared, the output compare register 2Dxy (OCR2Dxy) to which TCNT2Dx is compared, one timer down counter Dxy (DCNTDxy), two upper-limit setting registers Dxy (CUCR1Dx, CUCR2Dx) for two counters, and two input capture registers (ICR1Dxy and ICR2Dxy).

One-shot pulse can be output from timer D. By using compare match A, compare match B, compare match A or B, (or compare match A and B) as a start trigger of a down-counter, one-shot pulse with an offset can also be output.

Setting the TDE bit in ATUENR and bit STRDx in timer start register D (TSTRD) to 1 makes two timer counters 1Dx, 2Dx (TCNT1Dx and TCNT2Dx) in subblock Dx start operation.

The down-counter is started by any of four (five) sources; setting the DSTDxy bit in the down count start register Dx (DSTRDx) to 1, compare match A, compare match B, compare match A or B, (and compare match A and B) that are set by the TRGSELDxy bit in the down counter control register Dx (DCRDx). When any of these triggers occurs, DCNTDxy driven by the down counter clock is started.

The down counter is stopped on an underflow of the down counter and compare match B or A or B selected by bit TRGSELxy. The down counter is stopped immediately after it underflows (the counter is to be decremented when the value is $0000\ 0000_H$). It is stopped and cleared to $0000\ 0000_H$ on the first edge of the down counter clock after the trigger is detected.

For compare match A between TCNT1Dx and OCR1Dxy and compare match B between TCNT2Dx and OCR2Dxy are set to 1 on the first edge of the PCLK after the compare match. A compare match interrupt request is issued to CPU triggered by the compare match. According to the settings of range comparison value setting registers (RCR1Dx and RCR2Dx), a compare match A between OCR1Dxy and TCNT1Dx can be generated if the TCNT1Dx value is within the range of OCR1Dxy ≤ TCNT1Dx ≤ (OCR1Dxy + the value of the range specified by RCR1Dx) when writing to OCR1Dxy, and a compare match B between OCR2Dxy and TCNT2Dx can be generated if the TCNT2Dx value is within the range of OCR2Dxy ≤ TCNT2Dx ≤ (OCR2Dxy + the value of the range specified by RCR2Dx) when writing to OCD2Dxy.

Furthermore, the counter value of TCNT2Dx is captured into ICR2Dxy triggered by compare match A and the counter value of TCNT1Dx is capture into ICR1Dxy triggered by compare match B. The CMFADxy and CMFBDxy bits in TSRDx are set to 1 in synchronization with PCLK in the occurrence of compare match.

A signal on pin TODxyA is output when the output source selected in the OSSDxy bit in timer I/O control register 1Dx (TIOR1Dx) is activated. For example, assume that compare match A is selected. A signal level set by the IOAxy bit is output on pin TODxyA on the first edge of the PCLK after the compare match between TCNT1Dx and OCR1Dxy.

Output of the one-shot pulse is synchronized with the down counter clock in a way similar to the down counter operation. The four (five) counter starting sources are synchronized with the PCLK, a signal on pin TODxyB is output on the first edge of the down counter clock after the source is activated. Underflow of the down counter, which is a counter stopping source, is synchronized with the down counter clock and compare match B and compare match A or B is synchronized with the PCLK. As to negation timing, the TODxyB signal is negated in synchronization with underflow and on the first edge of the down counter clock after the compare match B or A or B.

Output Pin	Assertion Timing	Negation Timing	Initial Value
TODxyB	On the first edge of the down counter clock after the counter starting source is activated	On the first edge of the down counter clock after compare match or on DCNTDxy underflow	0 (inverted depending on TOCRDx)

Note: If an assertion and a negation occur simultaneously, the negation takes priority.

The initial values output on pins TODxyA and TODxyB are 0. However, the output level can be inverted by setting timer output control register Dx (TOCRDx). Setting bit TONEADx to 1 makes pin TODxyA in subblock Dn inverted and setting bit TONEBDx to 1 makes pin TODxyB in subblock Dx inverted. By setting the output select register Dx (OSELRDx), the value of the corresponding output value register Dx (ODRDx) can be output to pins (TODxyA and TODxyB).

To set the clocks for TCNT1Dx, TCNT2Dx, or DCNTDxy or values in registers such as DCNTDxy, OCR1Dxy, and OCR2Dxy while TCNT1Dx or TCNT2Dx is in operation, note that the value to be set may lead to malfunction. For example, while setting the compare match value, the counter value may exceed the value to be set.

Figure 21.46 shows an operation example of one-shot pulse output for channel 0 in subblock D0.

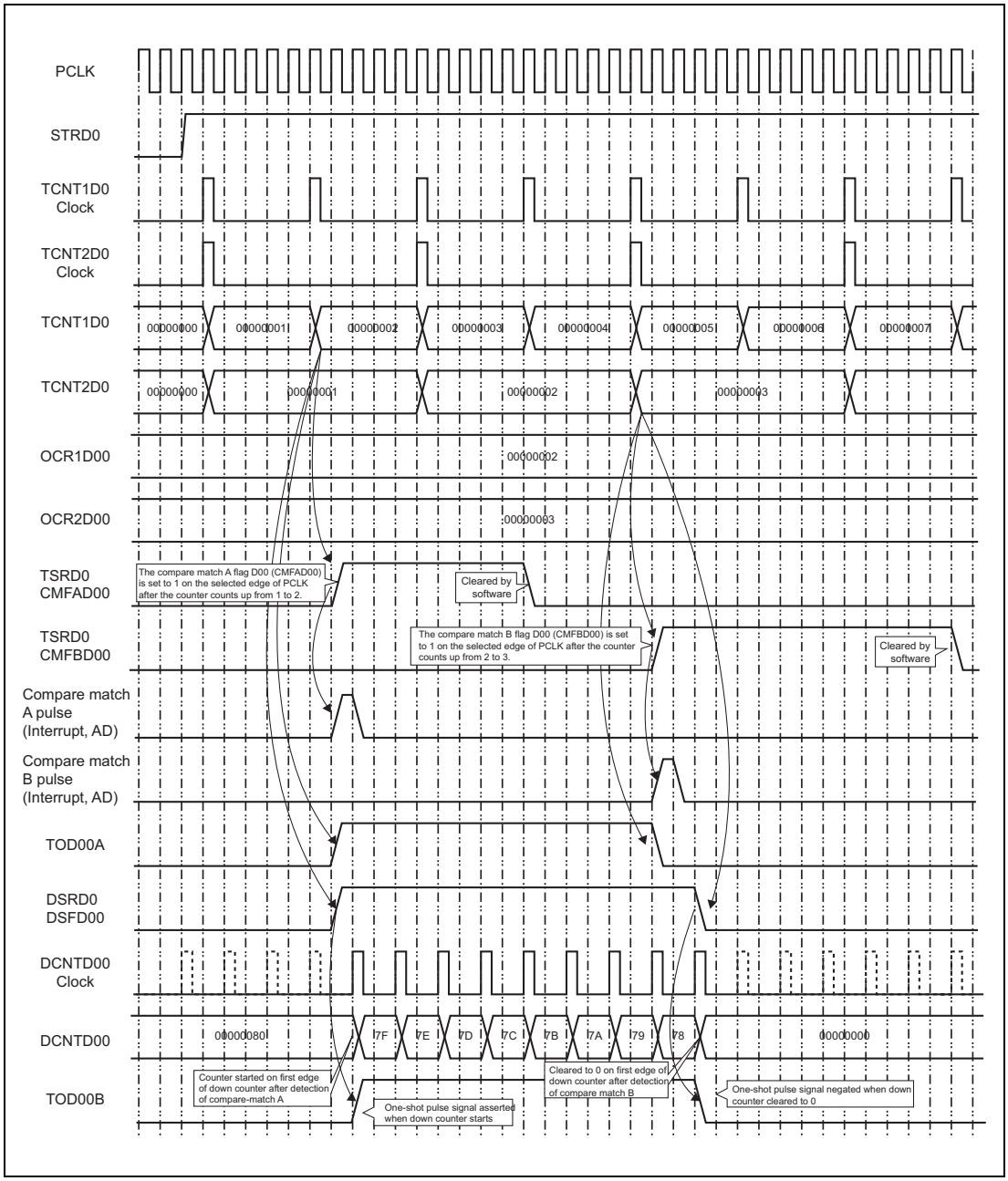


Figure 21.46 Operation Example of One-Shot Pulse Output (1) (Counting Started on Compare Match A and Stopped on Compare Match B)

Figure 21.46 shows operations of the down counter when the counter is started on compare match A and is stopped on compare match B. It also shows the assertion and negation of a one-shot pulse. Compare match A as the counter starting trigger and compare match B as the counter stopping trigger are set by the TRGSELD00 bits in DCRD0. Both matches are selected as the source of the output signal by the OSSD00 bits in TIOR1D0 and output levels are set by the IOAD00 and IOBD00 bits. A logical one for compare match A and a logical zero for compare match A are selected.

TCNT1D0 and TCNT2D0 are started for counting up on the first edge of the counter clock after the counter Dn start bit in timer start register D (TSTRD) is set to 1. If TCNT1D0 coincides with the output compare register (OCR1D00), compare match A occurs at the next PCLK cycle. At this time, the counter value of TCNT2Dx is captured in ICR2Dxy. In addition, a compare match interrupt request is issued to CPU triggered by compare match A. The compare match A flag D00 (CMFAD00) is set to 1.

At the same time, TOD00A outputs 1 and the down count status flag (DSFD00) is also set to 1, entering the down counter enabled state. This down count enabled state continues until detection of compare match B or under flow of DCNTD00. Input of the down counter clock during this state decrements DCNTD00. TOD00B outputs 1 from the time of the first down counter clock input.

If TCNT2D0 coincides with the output compare register (OCR2D00), compare match B occurs at the next PCLK cycle. At this time, the counter value of TCNT1Dx is captured into ICR1Dxy. In addition, a compare match interrupt request is issued to CPU triggered by compare match B. The compare match B flag D00 (CMFBD00) is set to 1. At this time, a level of 0 is output on pin TOD00A. The down counter is cleared and a one-shot pulse (TOD00B) is terminated.

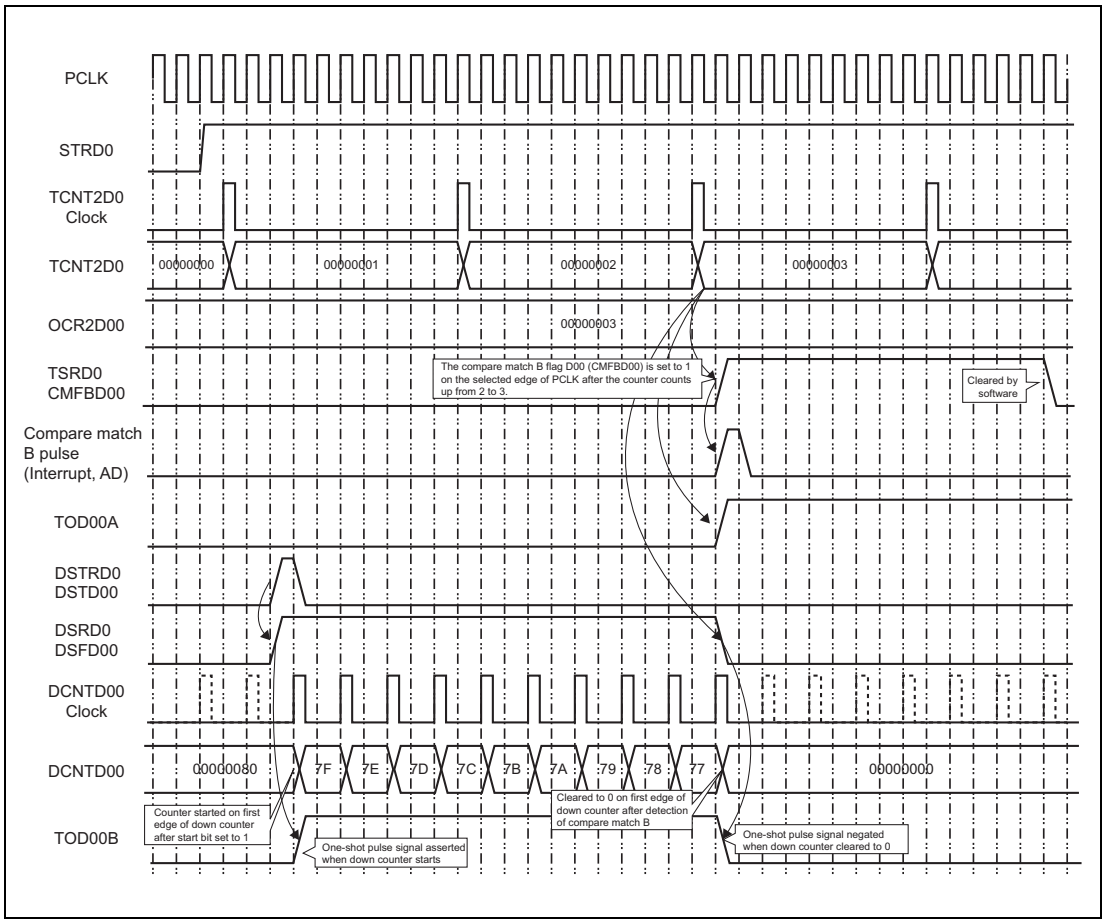


Figure 21.47 Operation Example of One-Shot Pulse Output (2) (Counting Started by Writing 1 to Counter Starting Bit and Stopped on Compare Match B)

Figure 21.47 shows an operation example when the down counter is started by writing 1 to the down counter starting bit. In this example, the counter starting trigger is not selected and compare match B is selected as the counter stopping trigger (TRGSELD00 in DCRD0). The source of the output signal is compare match B (OSSD00 in TIOR1D0) and logical one is output (IOBD00 in TIOR1D0).

Figure 21.48 shows an operation example of a one-shot pulse output for channel 0 in subblock D0. Setting the DSTD00 bit in the down counter starting register D0 (DSTRD0) sets the down counter status flag (DSFD00) to 1. This makes the down counter ready for counting down. DCNTD00 is started on the first edge of the down counter clock after DSFD00 is set to 1. At this time, a level of 1 is output on pin TOD00B.

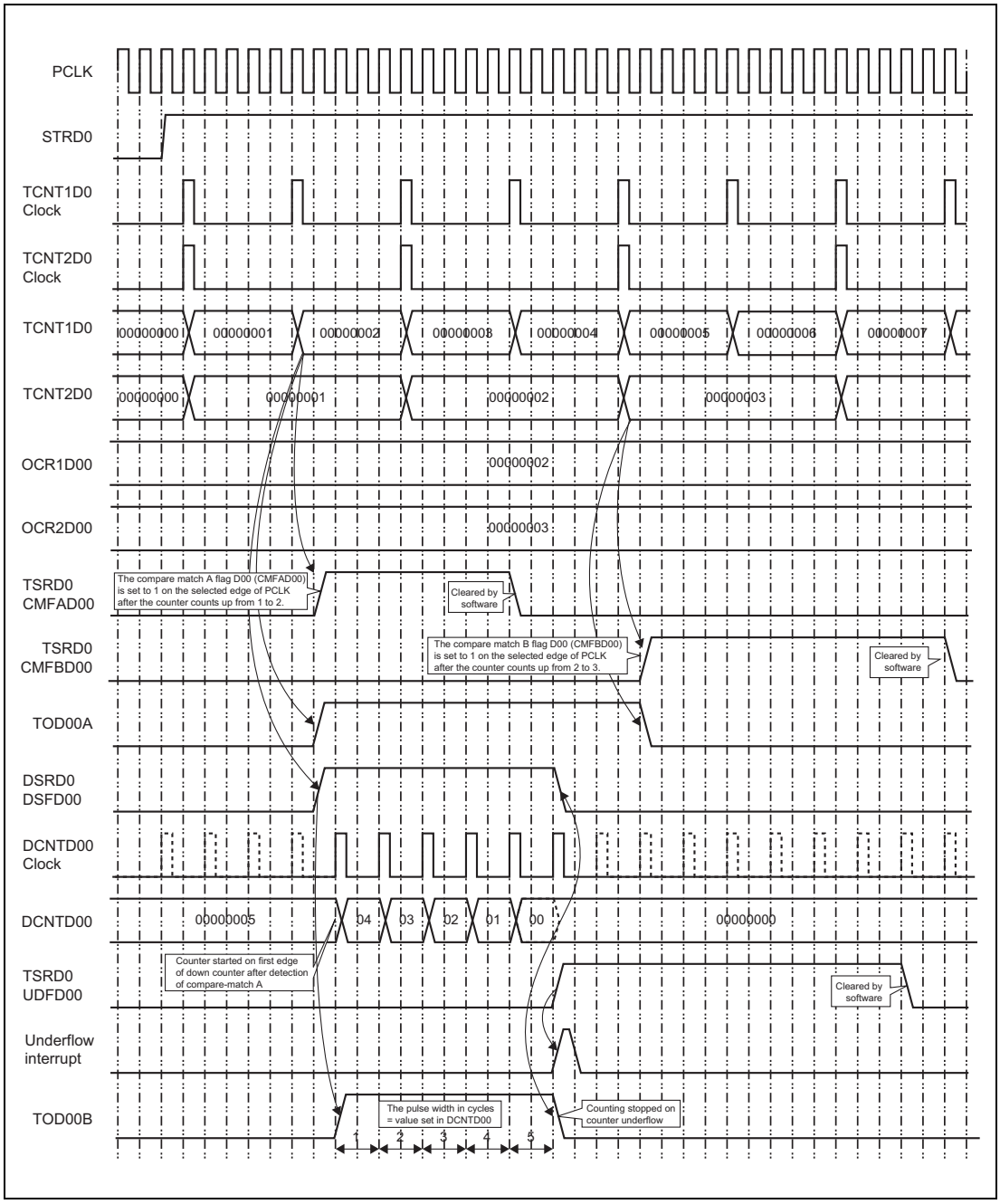


Figure 21.48 Operation Example of One-Shot Pulse Output (3) (Underflow Occurs)

Figure 21.48 shows an operation example when a one-shot pulse is terminated on underflow. In this example, compare match A as the counter starting trigger is selected and the counter stopping trigger is not selected (TRGSELD00 in DCRD0). Both compare matches A and B as the source of the output signal (OSSD00 in TIOR1D0) are selected. Output levels of 1 for compare match A (IOAD00) and 0 for compare match B (IOBD00) are selected.

Underflow is detected and the underflow flag D00 (UDFD00) in TSRD0 is set on the first edge of the down counter clock after the value in timer down counter Dxy (DCNTD00) is 000 000_H. At the same time, the one-shot pulse output is terminated. The width of the pulse output on pin TOD00B is equal to the value set in DCNTD00 before counting down.

21.7.3.1 Range Comparison Function

Figure 21.49 shows the operation of compare match occurrence using the range compare function according to the setting of the range comparison value setting register 1Dx (RCR1Dx).

If TCNT1Dx is larger than the output compare register 1Dxy (OCR1Dxy) value when writing to OCR1Dxy, a compare match A between OCR1Dxy and TCNT1Dx can be generated if the TCNT1Dx value is within the range of OCR1Dxy ≤ TCNT1Dx ≤ (OCR1Dxy + the value of the range specified by RCR1Dx).

At this time, the TCNT1Dx value is captured into OCR1Dxy.

The range comparison value setting register 2Dx (RCR2Dx) behaves similarly to TCNT2Dx.

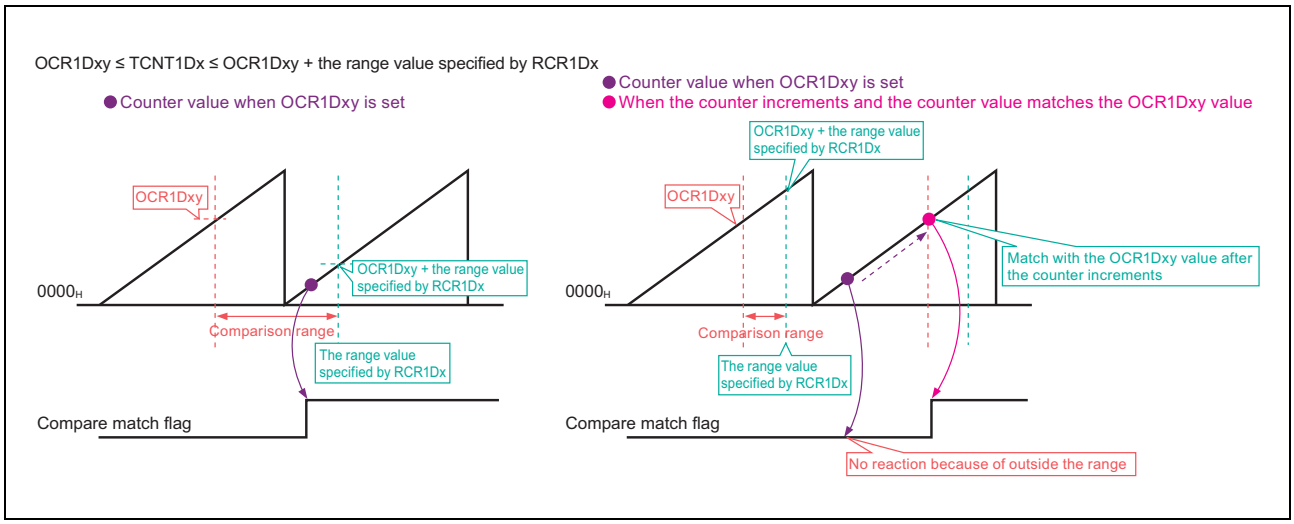


Figure 21.49 Operation of the Range Comparison Function

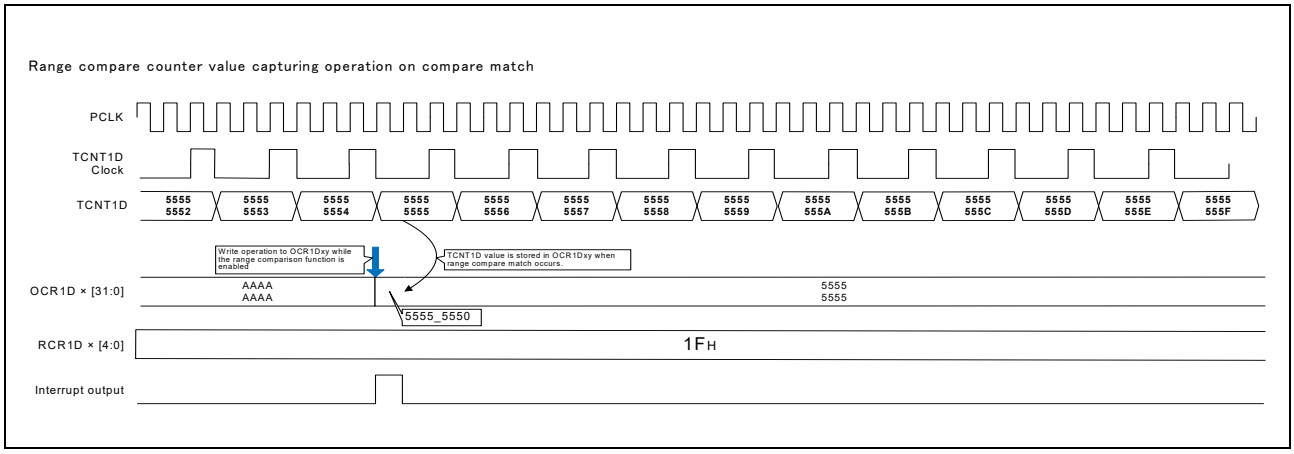


Figure 21.50 Range Compare Counter Value Capturing Operation

21.7.3.2 Counter Upper-Limit Setting Function

According to the setting for the counter upper-limit setting compare register 1Dx (CUCR1Dx), the upper-limit value of the timer counter 1Dx (TCNT1Dx) in each subblock can be modified to change the one-shot pulse cycle. To enable the counter upper-limit setting function, the CLR1Dx bit in the timer control register Dx (TCRDx) must be set to 1.

According to the setting for the counter upper-limit setting compare register 2Dx (CUCR2Dx), the upper-limit value of the timer counter 2Dx (TCNT2Dx) in each subblock can be modified to change the one-shot pulse cycle. To enable the counter upper-limit setting function, the CLR1Dx bit in the timer control register Dx (TCRDx) must be set to 1.

Figure 21.51 shows an example of operation when the counter upper-limit setting function in block D1 is enabled.

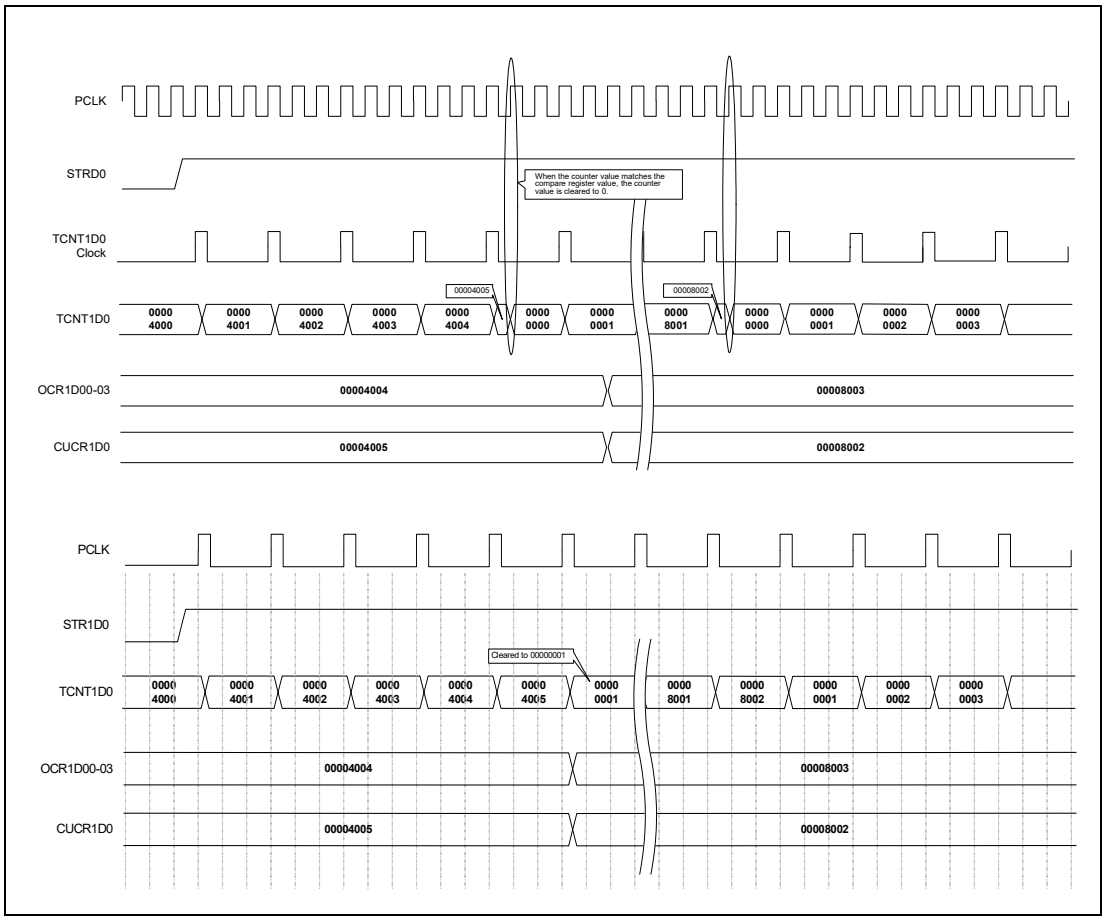


Figure 21.51 Operation of the Counter Upper-Limit Setting Function

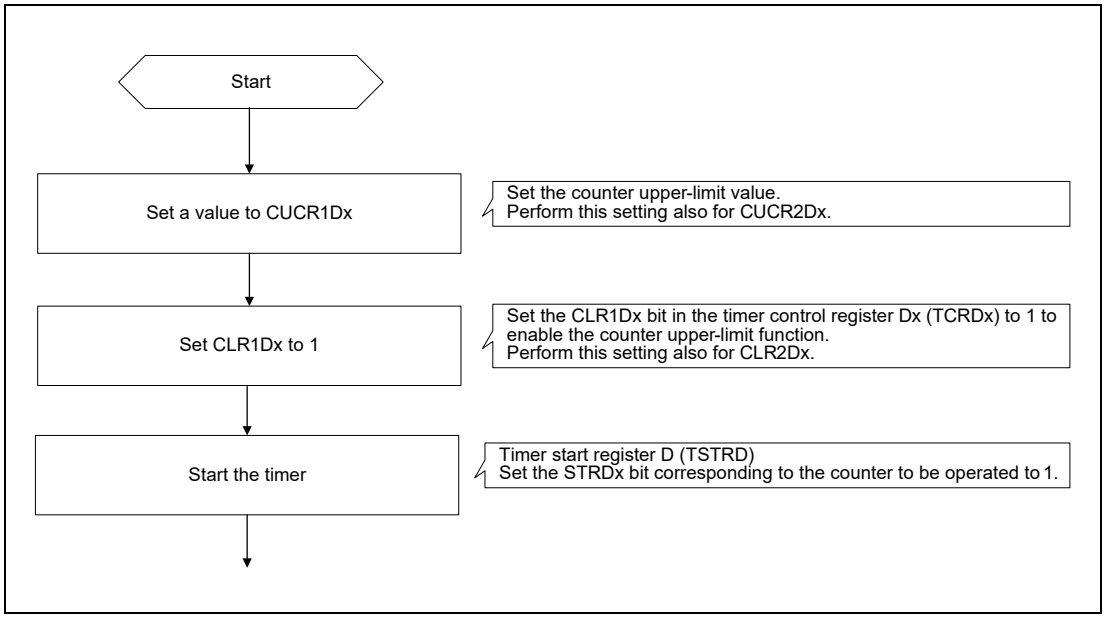


Figure 21.52 Counter Upper-Limit Function Setting Procedure

21.7.3.3 Capture Function

The input capture registers 1Dxy (ICR1Dxy) and 2Dxy (ICR2Dxy) capture the values counted by timer counters 1Dx (TCNT1Dx) and 2Dx (TCNT2Dx) in response to the compare match B and compare match A triggers, respectively. Capture proceeds on the rising edges of the pulses produced by matches.

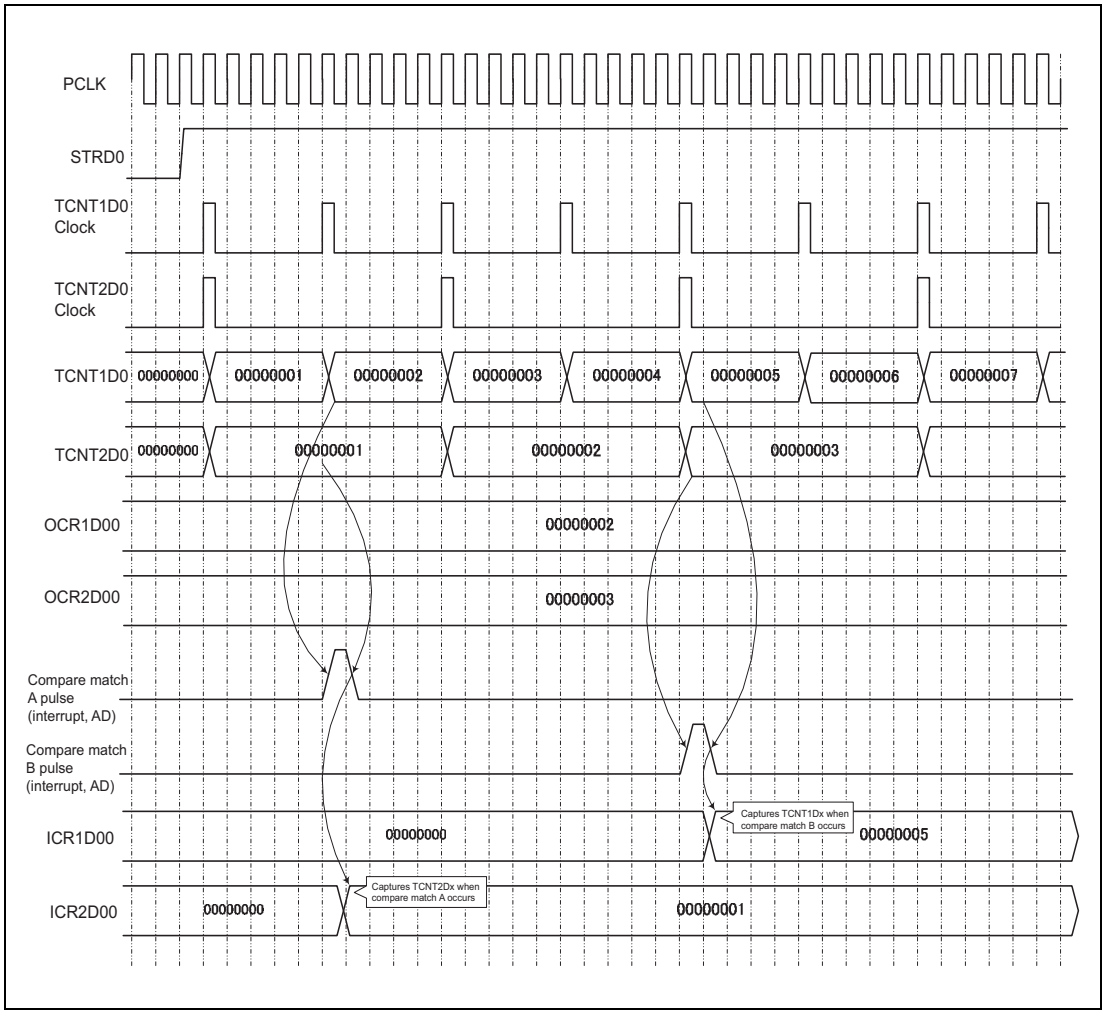


Figure 21.53 Capture Operation for Input Capture Registers 1Dxy and 2Dxy (ICR1Dxy, ICR2Dxy)

21.8 Timer E

21.8.1 Operation Overview

The timer E block is a PWM output timer that consists of seven timer E subblocks.

Timer E subblocks realize the following features:

- Output of waveform with a duty cycle of 0 to 100% by setting cycle-setting register and duty cycle-setting register
- The values of the cycle-setting and duty-cycle-setting registers are updated every PWM cycle. The values in the cycle reload register and duty cycle reload register are reloaded as update data. The reloading function can be enabled and disabled.
- Writing $000\ 000_H$ to the 24 higher-order bits of the counter can forcibly end PWM cycle.
- The shutoff input pins (POEx) shut off the PWM output. (They can also be used as the timer F input pins.) This feature is supported by subblocks E0 to E5. Subblock E6 does not support this feature.
- On-state duty (active-high output) and off-state duty (active-low output) modes available
- Interrupt requests can be issued on cycle match (compare match between cycle setting register and timer counter), that is, interrupts are issued every cycle.
- Cycle matches of channel 0 can be used as DMAC activation interrupts.
- Compare matches of the duty cycle setting register generate interrupt requests periodically.

A timer E subblock Ex consists of 4 channels. Components of each channel include a 24-bit timer counter (TCNTExy), a 24-bit cycle-setting register (CYLRExy), a 24-bit duty-cycle-setting register (DTRExy), a 24-bit cycle reload register Exy (CRLDExy), a 24-bit duty-load register (DRLDExy), and a controller.

Subblocks E0 to E5 has a shutoff input pin (POEx), and each channel of each subblock has a PWM output pin (TOExy). (The initial value of TOExy: 0)

POEx can also be used as TIF0A to TIF2A and TIF0B to TIF2B of timer F. For timer E, signals of TIF0A to TIF2A and TIF0B to TIF2B that have been processed by a noise canceler are used, which means that their noise canceler feature can be used. To use the noise canceler for the shutoff input signal of the PWM output shutoff feature, the timer F noise canceler setting is necessary as well as the timer E setting. For more about the setting, see **Section 21.9, Timer F**.

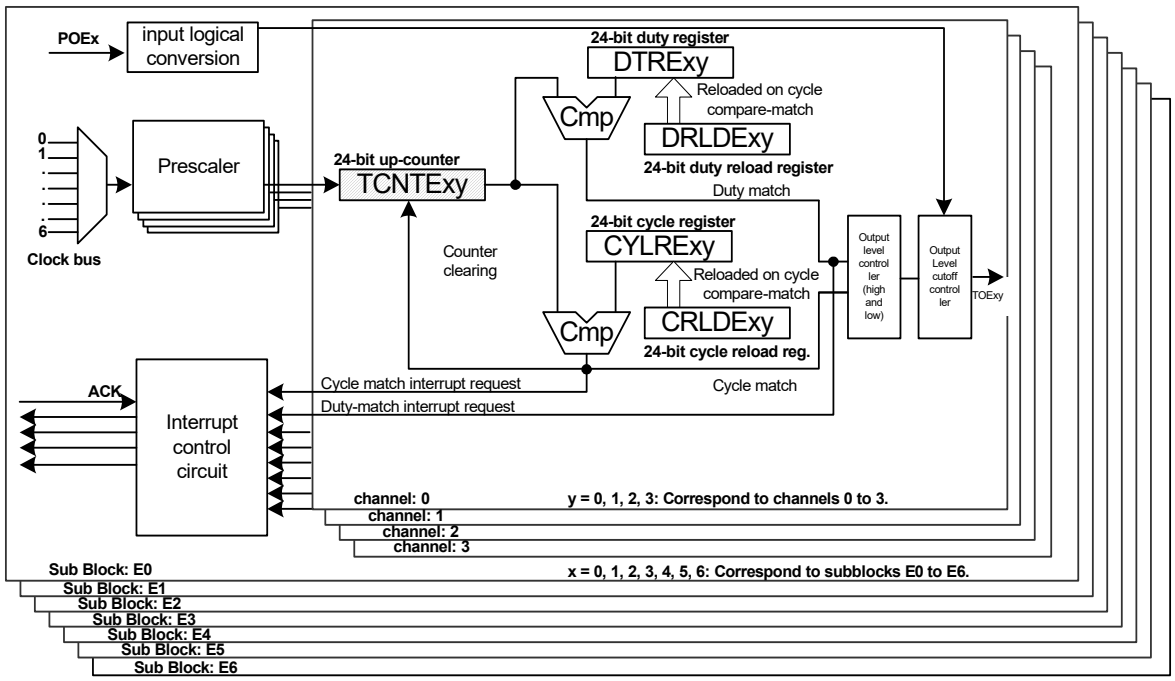
Table 21.96 shows the correspondence between TIF0A/B to TIF2A/B and POEx.

Table 21.96 Correspondence Between Timer F Primary Inputs and Timer E Shutoff Inputs

Timer F Primary Inputs	Timer E Shutoff Inputs
TIF0A	POE0
TIF1A	POE1
TIF2A	POE2
TIF0B	POE3
TIF1B	POE4
TIF2B	POE5

Be noted that the timer F measurement signals become the shutoff input signals when TIF0A/B to TIF2A/B are used as POEx inputs and the corresponding timer F subblocks are operated simultaneously.

Figure 21.54 is a block diagram of timer E.



Note: Interrupt requests to general DMA is supported only for channel 0 of each subblock. POEx input is supported only for subblock 0 to 5.

Figure 21.54 Block Diagram of Timer E

21.8.2 Registers Related to Timer E

21.8.2.1 TSTRE — Timer Start Register E

Access: 8-bit access is possible.

Address: FFE6 2000_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	STRE6	STRE5	STRE4	STRE3	STRE2	STRE1	STRE0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.97 TSTRE Register Contents

Bit Position	Bit Name	Function
7	—	Not used. This bit is always read as 0. This bit is read as 0. When writing, always write 0.
6	STRE6	Subblock E6 Start 0: Subblock E6 is disabled 1: Subblock E6 is enabled
5	STRE5	Subblock E5 Start 0: Subblock E5 is disabled 1: Subblock E5 is enabled
4	STRE4	Subblock E4 Start 0: Subblock E4 is disabled 1: Subblock E4 is enabled
3	STRE3	Subblock E3 Start 0: Subblock E3 is disabled 1: Subblock E3 is enabled
2	STRE2	Subblock E2 Start 0: Subblock E2 is disabled 1: Subblock E2 is enabled
1	STRE1	Subblock E1 Start 0: Subblock E1 is disabled 1: Subblock E1 is enabled
0	STRE0	Subblock E0 Start 0: Subblock E0 is disabled 1: Subblock E0 is enabled

TSTRE controls subblocks E0 to E6.

The timer E counters run when the timer E enable bit (TEE) in the ATU-IV master enable register (ATUENR), timer start register E (TSTRE), and subblock starting register E (SSTRE) must be set.

When this register is reset, it is initialized to 00_H.

CAUTION

To synchronize timer operation for all channels in the subblock, do not use the SSTREx register; instead, use the TSTRE register to enable or disable operation for each subblock.

21.8.2.2 SSTREx — Subblock Starting Registers Ex

(x = 0, 1, 2, 3, 4, 5, 6: Corresponding to subblocks E0 to E6, y = 0, 1, 2, 3: Corresponding to channels 0 to 3)

Access: 8-bit access is possible.

Address: FFE6 2100_H + (100_H * x)

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	SSTREx3	SSTREx2	SSTREx1	SSTREx0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 21.98 SSTREx Register Contents

Bit Position	Bit Name	Function
7 to 4	—	Not used. These bits are always read as 0. These bits are read as 0. When writing, always write 0.
3	SSTREx3	Counter Ex3 Start 0: Counter of channel 3 in subblock Ex is disabled 1: Counter of channel 3 in subblock Ex is enabled
2	SSTREx2	Counter Ex2 Start 0: Counter of channel 2 in subblock Ex is disabled 1: Counter of channel 2 in subblock Ex is enabled
1	SSTREx1	Counter Ex1 Start 0: Counter of channel 1 in subblock Ex is disabled 1: Counter of channel 1 in subblock Ex is enabled
0	SSTREx0	Counter Ex0 Start 0: Counter of channel 0 in subblock Ex is disabled 1: Counter of channel 0 in subblock Ex is enabled

The subblock starting registers E0 to E6 (SSTRE0 to SSTRE6) are 8-bit readable/writable registers. These registers control stop/start of timer counters of 4 channels of each subblock. This control is effective only for subblocks enabled by the timer start register E (TSTRE).

When SSTRE0 to SSTRE6 are reset, they are initialized to 00_H.

(1) SSTRExy — Counter Exy Start bit

Enables and disables timer counter Exy (TCNTExy).

SSTRExy	Function	
0	Counting of TCNTExy is disabled.	(Initial value)
1	Counting of TCNTExy is enabled.	

When this bit is cleared to 0, TCNTExy is disabled. TCNTExy retains the previous value while it is stopped. When this bit is set to 1 again, TCNTExy is resumed from the retained value.

Even if the counter Exy start bit is set to 1, counting does not start unless the master enable of timer Ex is set to 1 by the ATU-IV control register.

CAUTION

The prescalers run regardless of the counter Exy start bit and are not synchronized with the timing at which TCNTExy is started. Therefore, the time from when the counter Enm start bit is set to when TCNTExy is incremented for the first time is less than the cycle of the clock of TCNTE.

21.8.2.3 PSCREx — Prescaler Registers Ex

(x = 0, 1, 2, 3, 4, 5, 6: Correspond to subblocks E0 to E6.)

Access: 8-bit access is possible.

Address: FFE6 2104_H + (100_H * x)

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	PSCEx[2:0]		
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W

Table 21.99 PSCREx Register Contents

Bit Position	Bit Name	Function
7 to 3	—	Not used. Fix these bits to 0.
2 to 0	PSCEx[2:0]	Division Ratio These bits store the division ratio of the prescaler.

Prescaler registers E are 8-bit readable/writable registers.

Each subblock of timer E has one prescaler that divides the frequency of the clock supplied via the clock bus. The register sets the division ratio of the prescalers.

When the value in prescaler register E (PSCRE) is changed, the prescaler updates the value on its underflow. Timer counter E (TCNTExy) (y = 0 to 3) in the same block is driven by the clock output from prescaler Ex.

The settable value in prescaler register E (PSCRE) ranges from 0_H to 7_H. The division ratio is given below.

$$\text{Division ratio of prescaler} = \frac{1}{\text{PSCEx[2:0]} + 1} \quad \text{(Settable value: 1/1 to 1/8)}$$

A duty cycle of 50% for the prescaler E output clock is not guaranteed. The high level width is equal to the cycle of the PCLK and a low level is output in the remaining cycle of the prescaler E output clock.

Prescaler Ex runs when the TEE bit in the ATU-IV master enable register (ATUENR) and the subblock Ex start bit (STREx) in timer start register E (TSTRE) are both set to 1.

21.8.2.4 PSCCRExy — Prescaler Channel Registers Exy

(x = 0, 1, 2, 3, 4, 5, 6: Corresponding to subblocks E0 to E6, y = 0, 1, 2, 3: Corresponding to channels 0 to 3)

Access: 8-bit access is possible.

Address: FFE6 2118_H + (100_H * x) + y

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	PSCCExy[7:0]							
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.100 PSCCRExy Register Contents

Bit Position	Bit Name	Function
7 to 0	PSCCExy[7:0]	Division Ratio These bits store the division ratio of the prescaler.

Prescaler channel registers Exy are 8-bit readable/writable registers.

Each channel of timer E has one prescaler that divides the frequency of the clock supplied via the clock bus. The register sets the division ratio of the prescalers.

When the value in prescaler channel register Exy is changed, the prescaler updates the value on its underflow. Timer counter E (TCNTExy) (y = 0 to 3) in the same channel is driven by the clock output from prescaler Exy.

The settable value in prescaler channel register Exy ranges from 00_H to FF_H. The division ratio is given below.

$$\text{Division ratio of prescaler} = \frac{1}{\text{PSCCExy[7:0]} + 1} \quad \text{(Settable value: 1/1 to 1/256)}$$

A duty cycle of 50% for the prescaler Exy output clock is not guaranteed. The high level width is equal to the cycle of the PCLK and a low level is output in the remaining cycle of the prescaler E output clock.

Prescaler Exy runs when the TEE bit in the ATU-IV timer enable register (ATUENR) and the subblock Ex start bit (STREx) in timer start register E (TSTRE) are both set to 1. The setting of the counter Exy start bit (SSTRExy) in the subblock start register Ex (SSTREx) does not affect operation of the prescaler Exy.

21.8.2.5 TCREx — Timer Control Register Ex

(x = 0, 1, 2, 3, 4, 5, 6: Corresponding to subblocks E0 to E6, y = 0, 1, 2, 3: Corresponding to channels 0 to 3)

Access: 8-bit access is possible.

Address: FFE6 2108_H + (100_H * x)

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	PSCSEL	—	—	—	—	CKSELEx[2:0]		
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R	R/W	R/W	R/W

Table 21.101 TCREx Register Contents

Bit Position	Bit Name	Function
7	PSCSEL	Prescaler Select 0: These bits select the prescaler registers Ex (PSCREx: 3-bit settings for subblock units). 1: These bits select the prescaler channel registers Exy (PSCRExy: 8 bit settings for channel units).
6 to 3	—	Not used. These bits are always read as 0. When writing, always write 0.
2 to 0	CKSELEx[2:0]	TCNTExy Clock Select These bits select the counter clock of prescaler E from clock-bus lines 0 to 6.

Timer control registers Ex (TCREx) are 8-bit readable/writable registers that select the counter clock of prescaler E from clock-bus lines 0 to 6. 16-bit timer counter Exy (TCNTExy) is driven by the clock output from prescaler E.

(1) PSCSEL — Prescaler Select

The prescaler can be selected in subblock units or in channel units.
This selection makes a difference in the range of division ratios that can be selected for the prescaler.

(2) CKSELEx[2:0] — Subblock Ex Clock Select

The subblock Ex clock select (CKSELEx) selects the count source of the prescaler of subblock Ex.

CKSELEx				
[2]	[1]	[0]	Function	
0	0	0	Clock-bus line 0 is selected as counter clock of prescaler E	(Initial value)
0	0	1	Clock-bus line 1 is selected as counter clock of prescaler E	
0	1	0	Clock-bus line 2 is selected as counter clock of prescaler E	
0	1	1	Clock-bus line 3 is selected as counter clock of prescaler E	
1	0	0	Clock-bus line 4 is selected as counter clock of prescaler E	
1	0	1	Clock-bus line 5 is selected as counter clock of prescaler E	
1	1	0	Clock-bus line 6 is selected as counter clock of prescaler E.	
1	1	1	Reserved	

21.8.2.6 RLDCREx — Reload Control Registers Ex

(x = 0, 1, 2, 3, 4, 5, 6: Corresponding to subblocks E0 to E6, y = 0, 1, 2, 3: Corresponding to channels 0 to 3)

Access: 8-bit access is possible.

Address: FFE6 $210A_H$ + (100_H * x)

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	RLDENEx3	RLDENEx2	RLDENEx1	RLDENEx0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 21.102 RLDCREx Register Contents

Bit Position	Bit Name	Function
7 to 4	—	Not used. These bits are always read as 0. When writing, always write 0.
3 to 0	RLDENEx3 to RLDENEx0	Reload Enable Ex3-Ex0 0: Reload function on cycle match is disabled 1: Reload function on cycle match is enabled

Reload control registers Ex (RLDCREx) are 8-bit readable/writable registers. RLDENExy enables and disables the reload function.

When RLDCREx is reset, it is initialized to 00_H.

(1) RLDENExy — Reload Enable Exy

If the reload feature is enabled by setting this bit to 1, the value of the cycle reload register Exy (CRLDExy) is set to the cycle-setting register, and the value of the duty reload register Exy (DRLDExy) is set to the duty cycle setting register at the time of cycle matches of the timer counter Exy (TCNTExy) and the cycle-setting register Exy (CYLRExy).

21.8.2.7 POECREx — Output Shutoff Control Register Ex

(x = 0, 1, 2, 3, 4, 5, 6: Corresponding to subblocks E0 to E6, y = 0, 1, 2, 3: Corresponding to channels 0 to 3)

Access: 16-bit access is supported. 8-bit access is prohibited.

Address: FFE6 $210C_H$ + (100_H * x)

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	POECRKEY[7:0]								—	—	—	POEPOLx	POEENx3	POEENx2	POEENx1	POEENx0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	(W)*1	(W)*1	(W)*1	(W)*1	(W)*1	(W)*1	(W)*1	(W)*1	R	R	R	R/W	R/W	R/W	R/W	R/W

Note 1. Values written to the write key code are not retained.

Table 21.103 POECREx Register Contents

Bit Position	Bit Name	Function
15 to 8	POECRKEY[7:0]	Write Key Code $9C_H$: Values of POEENxy and POEPOLx can be changed. Other than $9C_H$: Values of POEENxy and POEPOLx cannot be changed.
7 to 5	—	Not used. These bits are always read as 0. When writing, always write 0.
4	POEPOLx	Shutoff Input Active Level Selection 0: Shutoff input is L active. 1: Shutoff input is H active.
3	POEENx3	Output Shutoff Enable/Disable 0: Output shutoff is disabled. 1: Output shutoff is enabled.
2	POEENx2	
1	POEENx1	
0	POEENx0	

Output shutoff register Ex (POECREx) is a 16-bit readable/writable register. To write a value to this register, the data to be written must be provided and the write key code must be set in bits 15-8. This feature is not supported for subblock 6 because there is no POE6 input. Do not enable the output shutoff enable/disable bits (POEEN60 to POEEN63) for POECRE6.

POECREx is initialized to 0000_H by a reset.

(1) POECRKEY — Write Key Code

This code controls the write permission for bits POEENxy (y = 0 to 3) and POEPOLx. Setting is required to bits 15-8 as well as the data to be written to bits POEENxy (y = 0 to 3) and POEPOLx. Data written to these bits is not retained; Value 00_H is always returned when being read.

(2) POEPOLx — Shutoff Input Active Level Selection

Controls the active level of the shutoff input (POEx). The setting of this bit applies to all channels of the subblock.

(3) POEENxy — Output Shutoff Enable/Disable

Controls disable/enable of the output shutoff feature. This bit can be set for each channel.

21.8.2.8 SOLVLEx — Output Shutoff Level Setting Register Ex

(x = 0, 1, 2, 3, 4, 5, 6: Corresponding to subblocks E0 to E6)

Access: 8-bit access is possible.

Address: FFE6 $210E_H$ + (100_H * x)

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	PWMSLVx3	PWMSLVx2	PWMSLVx1	PWMSLVx0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 21.104 SOLVLEx Register Contents

Bit Position	Bit Name	Function
7 to 4	—	Not used. These bits are always read as 0. When writing, always write 0.
3	PWMSLVx3	Output Level H/L Selection 0: L level is output to TOEx when output is shutoff. 1: H level is output to TOEx when output is shutoff.
2	PWMSLVx2	
1	PWMSLVx1	
0	PWMSLVx0	

The output shutoff level setting register Ex (SOLVLEx) is an 8-bit readable/writable register. When SOLVLEx is reset, it is initialized to 00_H.

(1) PWMSLVxy — Output Level H/L Selection

This bit specifies the output level of TOExy in the output shutoff state.

21.8.2.9 TSREx — Timer Status Registers E

(x = 0, 1, 2, 3, 4, 5, 6: Corresponding to subblocks E0 to E6, y = 0, 1, 2, 3: Corresponding to channels 0 to 3)

Access: 8-bit access and 16-bit access are possible.

Address: FFE6 2110_H + (100_H * x)

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	DMF Ex3	DMF Ex2	DMF Ex1	DMF Ex0	OVF Ex3	OVF Ex2	OVF Ex1	OVF Ex0	CMF Ex3	CMF Ex2	CMF Ex1	CMF Ex0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 21.105 TSREx Register Contents

Bit Position	Bit Name	Function
15 to 12	—	Not used. These bits are always read as 0. When writing, always write 0.
11 to 8	DMFEx3-DMFEx0	Duty Match Flags Ex3-Ex0 0: No duty match has occurred. 1: A Duty match has occurred.
7 to 4	OVFEx3-OVFEx0	Overflow Flags Ex3-Ex0 0: Counter E has not overflowed 1: Counter E has overflowed
3 to 0	CMFEx3-CMFEx0	Cycle Match Flags Ex3-Ex0 0: Cycle match has not occurred 1: Cycle match has occurred

The timer status register Ex (TSREx) is a 16-bit read-only register. In subblocks E0 to E6, these registers indicate occurrence of compare match (cycle match) of cycle-setting registers Exy (CYLRExy) and timer counters Exy (TCNTExy), compare match (duty match) of duty cycle setting registers Exy (DTRExy) and timer counters Exy (TCNTExy), and overflow of TCNTExy.

The overflow flag is the status flag for indicating the occurrence of overflow. It does not generate interrupts. The cycle match flag and the duty match flag are status flags for requesting interrupts. Interrupts can be issued by setting the corresponding bits of the TIEREx register. Setting the corresponding flags issues interrupt requests. To cancel an interrupt request, clear this flag by the timer status clear register Ex (TSCREx). Setting a bit of the timer status clear register Ex (TSCREx) clears the corresponding flag. Cycle matches of channel 0 can issue transfer requests to DMAC. DMA transfer requests are issued upon cycle matches regardless of the setting of this bit.

When TSREx is reset, it is initialized to 0000_H.

(1) DMFExy — Duty Match Flag Exy

DMFExy	Function
0	[Clearing condition] When 1 is written to the DMFCExy bit of the timer status clear register Ex (TSCREx). (Initial value)
1	[Setting condition] When the counter clock is input in the condition where the timer counter Exy (TCNTExy) value coincides with the duty cycle setting register Exy (DTRExy) value.

This flag cannot be set to 1 or 0 by software. The next duty match can be processed even if DMFExy is 1 (even if the flag is not cleared). 1 is overwritten to these bits.

(2) CMFExy — Cycle Match Flag Exy

CMFExy	Function
0	[Clearing conditions] When 1 is written to the CMFCExy bit of the timer status clear register Ex (TSCREx). (Initial value)
1	[Setting condition] When timer counter Exy (TCNTExy) is incremented while it is the same value as cycle setting register Exy (CYLRExy)

These bits cannot be set to 1 or 0 by software. Even if these bits are 1, meaning that the flag has not been cleared, the next cycle match and DMA transfer request can be input. In this case, 1 is rewritten to these bits.

(3) OVFExy — Overflow Flag Exy

OVFExy	Function
0	[Clearing condition] When 1 is written to the OVFCExy bit of the timer status clear register Ex (TSCREx). (Initial value)
1	[Setting condition] When the 24 higher-order bits of the timer counter Exy (TCNTExy) value changes from FFF FFF_H to 000 000_H

This flag cannot be set to 1 or 0 by software. The next cycle match and DMA transfer can work even if CMFExy is 1 (even if the flag is not cleared). 1 is overwritten to these bits.

These bits are set to 1 when timer counter Exy (TCNTExy) overflows. This flag cannot be set to 1 or 0 by software.

Overflow occurs when the 24 higher-order bits of the timer counter Exy is incremented while it is FFF FFF_H. Writing 000 000_H to the 24 higher-order bits of the timer counter Exy has no effect on these bits.

When writing a value to the 24 higher-order bits of the timer counter and incrementation occur simultaneously while the counter is FFF FFF_H, the overflow flag is set to 1 but the counter value is changed to the written value instead of 000000_H.

No interrupt corresponds to these flags. As to cycle match, since the 24 higher-order bits of the counter is cleared to 000 001_H, overflow will not occur. However, it may occur when the value in the cycle setting register is changed during counter in operation.

When overflow and cycle match occur simultaneously, overflow is not detected (the 24 higher-order bits of the counter is incremented while it is FFF FFF_H and the 24 higher-order bits of CYLRExy is FFF FFF_H). In this case, only the cycle match is handled. If the 24 higher-order bits of the counter value is 000 001_H and the reload function is enabled, cycle reload or duty-cycle reload is performed. The above operation applies for the case when an overflow and a duty match occur simultaneously.

21.8.2.10 TSCREx — Timer Status Clear Register Ex

(x = 0, 1, 2, 3, 4, 5, 6: Corresponding to subblocks E0 to E6, y = 0, 1, 2, 3: Corresponding to channels 0 to 3)

Access: 8-bit access and 16-bit access are possible.
Value 0 cannot be written. The value is not retained when 1 is written. These bits are always read as 0.

Address: FFE6 2112_H + (100_H * x)

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	DMFCEx3	DMFCEx2	DMFCEx1	DMFCEx0	OVFCEx3	OVFCEx2	OVFCEx1	OVFCEx0	CMFCEx3	CMFCEx2	CMFCEx1	CMFCEx0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.106 TSCREx Register Contents

Bit Position	Bit Name	Function
15 to 12	—	Not used. These bits are always read as 0. When writing, always write 0.
11 to 8	DMFCExy	Duty Match Flag Clear Enable Exy 0: Disabled (Initial value) 1: Clears DMFExy of timer status register Ex (TSREx) to 0.
7 to 4	OVFCExy	Overflow Flag Clear Enable Exy 0: Disabled (Initial value) 1: Clears OVFExy of timer status register Ex (TSREx) to 0.
3 to 0	CMFCExy	Cycle Match Flag Clear Enable Exy 0: Disabled (Initial value) 1: Clears CMFExy of timer status register Ex (TSREx) to 0.

The timer status clear register Ex (TSCREx) is a 16-bit readable/writable register. This register specifies flag clearing in the events of overflow of timer counters Exy (TCNTExy), cycle match of TCNTExy and cycle-setting registers Exy (CYLRExy), and duty match of TCNTExy and duty cycle setting registers Exy (DTRExy).

TSCREx is readable/writable in 16-bit units. Value 0 is always returned when being read.

When TSCREx is reset, it is initialized to 0000_H.

(1) DMFCExy — Duty Match Flag Clear Enable Exy

This flag specifies flag clearing at the time of duty match of TCNTExy and the duty cycle setting register Exy (DTRExy).

Setting this bit enables clearing of the duty match flag (DMFExy) of the timer status register Ex (TSREx). This bit is always read as 0.

DMFCExy	Function
0	Disabled (Initial value)
1	Clears DMFExy of timer status register Ex (TSREx) to 0.

(2) OVFCExy — Overflow Flag Clear Enable Exy

This bit specifies flag clearing at the time of overflow of the timer counter Exy (TCNTExy).

Setting this bit enables clearing of the overflow flag Exy (OVFExy) of the timer status register Ex (TSREx). This bit is always read as 0.

OVFCExy	Function
0	Disabled (Initial value)
1	Clears OVFExy of timer status register Ex (TSREx) to 0.

(3) CMFCExy — Cycle Match Flag Clear Enable Exy

This flag specifies flag clearing at the time of cycle match of TCNTExy and the cycle-setting register Exy (CYLRExy).

Setting this bit enables clearing of the cycle match flag Exy (CMFExy) of the timer status register Ex (TSREx). This bit is always read as 0.

CMFCExy	Function
0	Disabled (Initial value)
1	Clears CMFExy of timer status register Ex (TSREx) to 0.

21.8.2.11 TIEREx — Timer Interrupt Enable Registers Ex

(x = 0, 1, 2, 3, 4, 5, 6: Correspond to subblocks E0 to E6, y = 0, 1, 2, 3: Corresponding to channels 0 to 3.)

Access: 8-bit access and 16-bit access are possible.

Address: FFE6 2116_H + (100_H * x)

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	DME Ex3	DME Ex2	DME Ex1	DME Ex0	—	—	—	—	CME Ex3	CME Ex2	CME Ex1	CME Ex0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 21.107 TIEREx Register Contents

Bit Position	Bit Name	Function
15 to 12	—	Not used. These bits are always read as 0. When writing, always write 0.
11 to 8	DMEEx3-DMEEx0	Duty match interrupt enable Ex3-Ex0 Specifies enable/disable of interrupt requests issued by duty match flags DMFExy.
7 to 4	—	Not used. These bits are always read as 0. When writing, always write 0.
3 to 0	CMEEx3-CMEEx0	Cycle Match Interrupt Enable Ex3-Ex0 Specifies enable/disable of interrupt requests issued by cycle match flags CMFExy.

The timer interrupt enable register Ex (TIEREx) is a 16-bit readable/writable register. This register controls enable/disable of interrupts triggered by compare match (cycle match) of the cycle-setting register Exy (CYLRExy) and TCNTExy and compare match (duty match) of the duty cycle setting register Exy (DTRExy) and TCNTExy.

When TIEREx is reset, it is initialized to 0000_H.

(1) DMEExy — Duty Match Interrupt Enable Exy

This flag specifies enable/disable of interrupt requests triggered by duty match of DTExy.

Setting this bit enables interrupts issued by the duty match flag (DMFExy) of the timer status register Ex (TSREx).

DMEExy	Function
0	Interrupts by DMFExy is disabled. (Initial value)
1	Interrupts by DMFExy is enabled.

(2) CMEExy — Cycle Match Interrupt Enable Exy

This flag specifies enable/disable of interrupt requests triggered by cycle matches of CYLExy.

CMEExy	Function
0	Interrupts by CMFExy is disabled.(Initial value)
1	Interrupts by CMFExy is enabled.

CAUTION

Interrupt requests by cycle match (CMFExy) and duty match (DMFExy) of timer subblock Ex are issued as the logical sum of CMFExy and DMFExy. Whether the interrupt request is triggered by a cycle match or a duty match of the counter can be determined by referencing TSREx. Clear the corresponding interrupt request flag of TSREx when processing the interrupt. The interrupt request will be repeated if the flag is not cleared.

21.8.2.12 TOCREx — Timer Output Control Registers Ex

(x = 0, 1, 2, 3, 4, 5, 6: Corresponding to subblocks E0 to E6, y = 0, 1, 2, 3: Corresponding to channels 0 to 3)

Access: 8-bit access is possible.

Address: FFE6 2114_H + (100_H * x)

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	TONEEx3	TONEEx2	TONEEx1	TONEEx0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 21.108 TOCREx Register Contents

Bit Position	Bit Name	Function
7 to 4	—	Not used. These bits are always read as 0. When writing, always write 0.
3 to 0	TONEExy	TOExy Output Inversion Select Inversion setting of the TOExy output signal (on-state duty/off-state duty selection)

Timer output control registers Ex (TOCRE0 to TOCRE6) are 8-bit readable/writable registers that select whether or not a signal on the output pin TOExy in each subblock Ex is inverted.

When this register is reset, it is initialized to 00_H.

(1) TONEExy — TOExy Output Inversion Select

These bits select whether or not a signal on the PWM output pin (TOE) is inverted.

TONEExy	function	
0	Normal output from the output pin (TOExy)	(Initial value)
1	Inverted output from the output pin (TOExy)	

The output signal is inverted on the first edge of the PCLK after duty modes are switched by the timer output control register. The operating state of the timer counter Exy (TCNTExy) has no effect on the mode switching.

The initial level on the PWM output pin is low (TONEExy = 0).

21.8.2.13 TCNTExy — Timer Counters Exy

(x = 0, 1, 2, 3, 4, 5, 6: Correspond to subblocks E0 to E6. y = 0, 1, 2, 3: Correspond to channels 0 to 3.)

Access: 32-bit access is supported. 8-bit access and 16-bit access are prohibited.

Address: FFE6 2124$_H$ + (100$_H$ * x) + (20$_H$ * y)

Value after reset: 0000 0100$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TCNTExy															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TCNTExy								—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Timer counters Exy (TCNTExy) are 24-bit readable/writable registers allocated in channel y of subblock Ex. When the 8 lower-order bits are read, 0 is always read. When writing to these bits, always write 0. These registers are started by setting the TEE bit in the ATU-IV master enable register (ATUENR), the subblock Ex start bit (STREx) in the timer start register E (TSTRE), and the counter Exy start bit (SSTRExy) in subblock starting register Ex (SSTREx).

The counter clock is selected by the TCNTEx clock select bits (CKSELEx[2:0]) in timer control register x (TCREx), and 2nd prescaler register Ex (PSCREx) of timer E.

The 24 higher-order bits of these counters are initialized to 000 001$_H$ on cycle match with cycle setting register Exy (CYLRExy). For example, when the value of the 24 higher-order bits in the cycle setting register is N and the 24 higher-order bits of the counter value is to be incremented from N to N + 1, the counter value is changed to 1. This enables counting from 1 to N and PWM pulses with the cycle time of N is produced.

The 24 higher-order bits of these counters can count from 000 001$_H$ to FFF FFF$_H$ (when the value of the 24 higher-order bits of the cycle setting register is FFF FFF$_H$).

When writing 000 000$_H$ to the 24 higher-order bits of these counters, a PWM cycle is terminated and a new PWM cycle is started in the next clock cycle. While the 24 higher-order bits of the counter value holds 000 000$_H$, the PWM output retains the previous value and outputs a level of 1 at the beginning of the new cycle. When the PWM cycle is terminated before duty cycle match, the duty cycle for that PWM cycle is 100% (1 is always output), that is, a level of 0 will not be output between PWM cycles. For details on writing 000 000$_H$ to the 24 higher-order bits of these counters, see **Figure 21.56** and **Figure 21.57**.

When TCNTExy or CYLRExy is rewritten during the counter in operation, a cycle match may not occur even if the value of the 24 higher-order bits of the counter reaches FFF FFF$_H$. In this case, the 24 higher-order bits of the counter value is changed from FFF FFF$_H$ to 000000$_H$ in the next counter clock cycle. A PWM cycle is terminated in a way similar to writing 0000$_H$. The 24 higher-order bits of the counter value is incremented to 000 001$_H$ and a new PWM cycle is started.

When the reload function is enabled, reloading of the cycle or duty cycle is also performed.

When the 24 higher-order bits of TCNTExy is reset, it is initialized to 000 001$_H$.

21.8.2.14 CYLRExy — Cycle-Setting Registers Exy

(x = 0, 1, 2, 3, 4, 5, 6: Correspond to subblocks E0 to E6. y = 0, 1, 2, 3: Correspond to channels 0 to 3.)

Access: 32-bit access is supported. 8-bit access and 16-bit access are prohibited.

Address: FFE6 2128$_H$ + (100$_H$ * x) + (20$_H$ * y)

Value after reset: FFFF FF00$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CYLRExy															
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CYLRExy								—	—	—	—	—	—	—	—
Value after reset	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

CYLRExy are 24-bit readable/writable registers that store the cycle of PWM. The settable value of the 24 higher-order bits ranges from 000 001$_H$ to FFF FFF$_H$.

When the 8 lower-order bits are read, 0 is always read. When writing to these bits, always write 0.

The value in CYLREnm is constantly compared with the value in the timer counter Exy (TCNTExy). When they match, the 24 higher-order bits of TCNTExy is cleared to 000 001$_H$. When the RLDENxy bit in the reload control register Ex (RLDCREx) is set to 1, the values in the cycle reload register Exy (CRLDExy) and duty cycle reload register Exy (DRLDExy) are transferred to cycle-setting register Exy (CYLRExy) and duty cycle setting register Exy (DTRExy). Writing 000 000$_H$ to the 24 higher-order bits of TCNTExy can forcibly end PWM cycle and start the same reloading as above at the next clock cycle.

Interrupt request to CPU and transfer request to DMAC (only channel 0) can be made by a compare match (cycle match) of the cycle-setting register. Do not make, however, an interrupt request and a transfer request to DMAC simultaneously.

To rewrite to CYLRExy during TCNTExy in operation, note that the value to be set may lead to malfunction. When TCNTExy in operation is rewritten, a cycle match between TCNTExy and DTRExy may not be detected and TCNTExy continues to be incremented even if the counter value exceeds the value in CYLRExy. In this case, unwanted PWM waveforms are output.

When the 24 higher-order bits of CYLRExy is reset, it is initialized to FFF FFF$_H$.

21.8.2.15 DTRExy — Duty Cycle Setting Registers Exy

(x = 0, 1, 2, 3, 4, 5, 6: Correspond to subblocks E0 to E6. y = 0, 1, 2, 3: Correspond to channels 0 to 3.)

Access: 32-bit access is supported. 8-bit access and 16-bit access are prohibited.

Address: FFE6 212C$_H$ + (100$_H$ * x) + (20$_H$ * y)

Value after reset: FFFF FF00$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DTREXy															
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DTREXy								—	—	—	—	—	—	—	—
Value after reset	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

DTREXy are 24-bit readable/writable registers that store the duty cycle of PWM. The settable value of the 24 higher-order bits ranges from 000 000$_H$ to FFF FFF$_H$.

When the 8 lower-order bits are read, 0 is always read. When writing to these bits, always write 0.

The value in DTREXy is constantly compared with the value in timer counter Exy (TCNTExy). When they match, the output level on the pin for the corresponding channel becomes 0. When the values in CYLREXy and TCNTExy match while the RLDENxy bit is set to 1, the value in DRLDExy is reloaded to DTREXy. Writing 000 000$_H$ to the 24 higher-order bits of TCNTExy can forcibly end PWM cycle and start the same reloading as above at the next clock cycle.

Interrupt requests to CPU can be issued by compare match (duty matches) of the duty cycle setting register.

The settable value of the 24 higher-order bits in DTREXy ranges from 0 to the value in CYLREXy. When 0 is set, the duty cycle is 0% and the same value as CYLREXy is set, the duty cycle is 100%. DTREXy must be set to the value less than CYLREXy.

To rewrite to DTREXy during TCNTExy in operation, note that the value to be set may lead to malfunction. When TCNTExy in operation is rewritten, a duty cycle match may not be detected. In this case, unwanted PWM waveforms may be output.

When the 24 higher-order bits of DTREXy is reset, it is initialized to FFF FFF$_H$.

21.8.2.16 CRLDExy — Cycle Reload Registers Exy

(x = 0, 1, 2, 3, 4, 5, 6: Correspond to subblocks E0 to E6. y = 0, 1, 2, 3: Correspond to channels 0 to 3.)

Access: 32-bit access is supported. 8-bit access and 16-bit access are prohibited.

Address: FFE6 2130_H + (100_H * x) + (20_H * y)

Value after reset: FFFF $FF00_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CRLDExy															
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CRLDExy								—	—	—	—	—	—	—	—
Value after reset	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Cycle reload registers Exy (CRLDExy) are 24-bit readable/writable register, and the 24 higher-order bits in this register can be set to 000 001_H to FFF FFF_H as the cycle of PWM outputs.

When the 8 lower-order bits are read, 0 is always read. When writing to these bits, always write 0.

When the reload function is enabled, the value in this register is transferred to the cycle-setting register Exy (CYLRExy) on cycle match.

Writing 000 000_H to the 24 higher-order bits of TCNTExy can forcibly end PWM cycle and start the same reloading as above at the next clock cycle.

When the 24 higher-order bits of CRLDExy is reset, it is initialized to FFF FFF_H.

21.8.2.17 DRLDExy — Duty Cycle Reload Registers Exy

(x = 0, 1, 2, 3, 4, 5, 6: Correspond to subblocks E0 to E6. y = 0, 1, 2, 3: Correspond to channels 0 to 3.)

Access: 32-bit access is supported. 8-bit access and 16-bit access are prohibited.

Address: FFE6 2134$_H$ (100$_H$ * x) + (20$_H$ * y)

Value after reset: FFFF FF00$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DRLDExy															
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DRLDExy								—	—	—	—	—	—	—	—
Value after reset	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Duty cycle reload registers Exy (DRLDExy) are a 24-bit readable/writable register, and the 24 high-order bits in this register can be set to 000 000$_H$ to FFF FFF$_H$ as the duty cycle.

When the 8 lower-order bits are read, 0 is always read. When writing to these bits, always write 0.

When the reload function is enabled, the value in this register is transferred to the duty cycle-setting register Exy (DTRExy) on cycle match.

Writing 000 000$_H$ to the 24 higher-order bits of TCNTExy can forcibly end PWM cycle and start the same reloading as above at the next clock cycle.

When the 24 higher-order bits of DRLDExy is reset, it is initialized to FFF FFF$_H$.

21.8.3 Operation Description

Timer E consists of 24-bit timer counter Exy (TCNTExy), 24-bit cycle-setting register Exy (CYLRExy), 24-bit duty cycle setting register Exy (DTRExy), 24-bit cycle reload register Exy (CRLDExy), and 24-bit duty cycle reload register Exy (DRLDExy). Timer E can be used as a PWM timer.

TCNTExy starts counting up when a channel is selected by subblock starting register Ex (SSTREx) after a subblock is selected by timer start register E (TSTRE). A logical zero level is output on pin TOExy on the first edge of the counter clock after TCNTExy matches duty cycle setting register Exy (DTRExy), or a logical one level is output on pin TOExy on the first edge the counter clock after TCNTExy matches cycle setting register Exy (CYLRExy). After a match with the cycle setting register, the 24 higher-order bits of the counter is set to $000\ 001_H$ on the next counter clock edge and starts counting up again.

Subsequently duty-cycle and cycle match are repeated, producing a PWM output on pin TOExy.

However, externally output level retains an initial value of 0 for one cycle which is from starting up the counter to the first cycle match.

The settable PWM cycle ranges from $000\ 001_H$ to $FFF\ FFF_H$. The settable duty cycle ranges from 0% to 100%. When the duty cycle setting register is set to $000\ 000_H$, the output level is 0 and remains unchanged (duty cycle = 0%). When the values in duty cycle setting register and cycle setting register are the same, the output level is 1 and remains unchanged (duty cycle = 100%). The value in duty cycle setting register must be equal to or less than the value in cycle setting register.

An interrupt request can be generated per cycle by a compare match (cycle match) of the cycle register. The cycle match of channel 0 can be used as a DMA transfer request. As shown in **Figure 21.55**, a DMA transfer request becomes 1 PCLK pulse output.

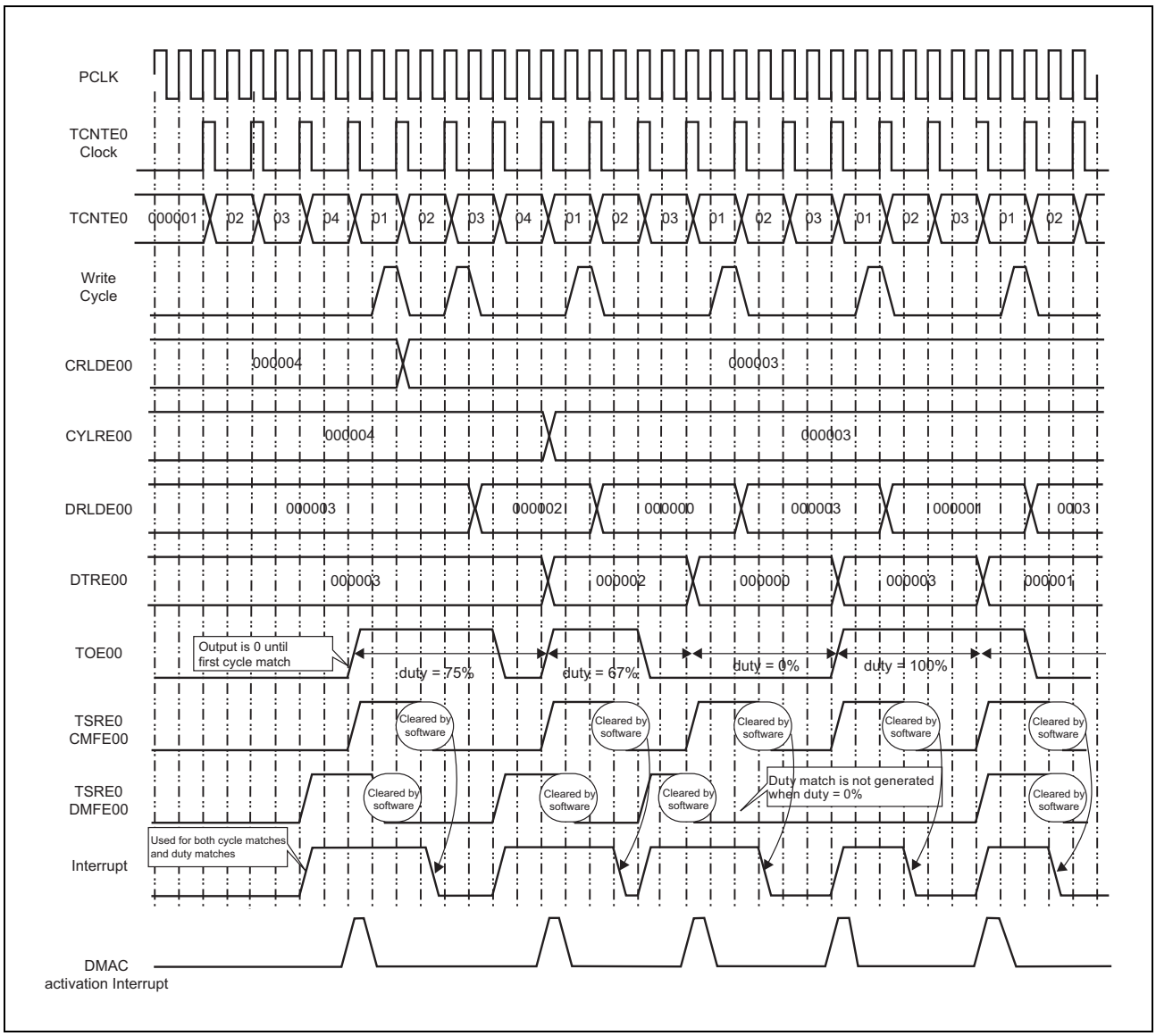


Figure 21.55 Operation of PWM (1)

The duty cycle setting and cycle setting registers have respective reload registers. When the values in the up-counter and cycle-setting register match, the values in the duty cycle reload and cycle reload registers are loaded to the duty cycle setting and cycle setting registers. The loaded data is updated in the next PWM cycle after loading. The reload function is enabled and disabled by the reload enable bit Exy (RLDENExy) in the reload control register Ex (RLDCREx).

In timer E, a PWM output cycle is terminated by writing $000\ 000_H$ to the 24 higher-order bits of the timer counter Exy (TCNTExy). The value of the 24 higher-order bits of the counter is changed from $000\ 000_H$ to $000\ 001_H$ on the next counter clock and the counter is restarted. When the 24 higher-order bits of the counter value is changed, the values in the duty cycle reload and cycle reload registers are loaded to the duty cycle setting and cycle setting registers.

Figure 21.55 illustrates the operation of the PWM timer of channel 0, block E0.

The figure illustrates an operation in which the duty is changed from 75% to 67%, 0%, and to 100% in every cycle.

Figure 21.56 shows the PWM output cycle terminated by writing 000 000_H to the 24 higher-order bits of the counter and the counter restarted.

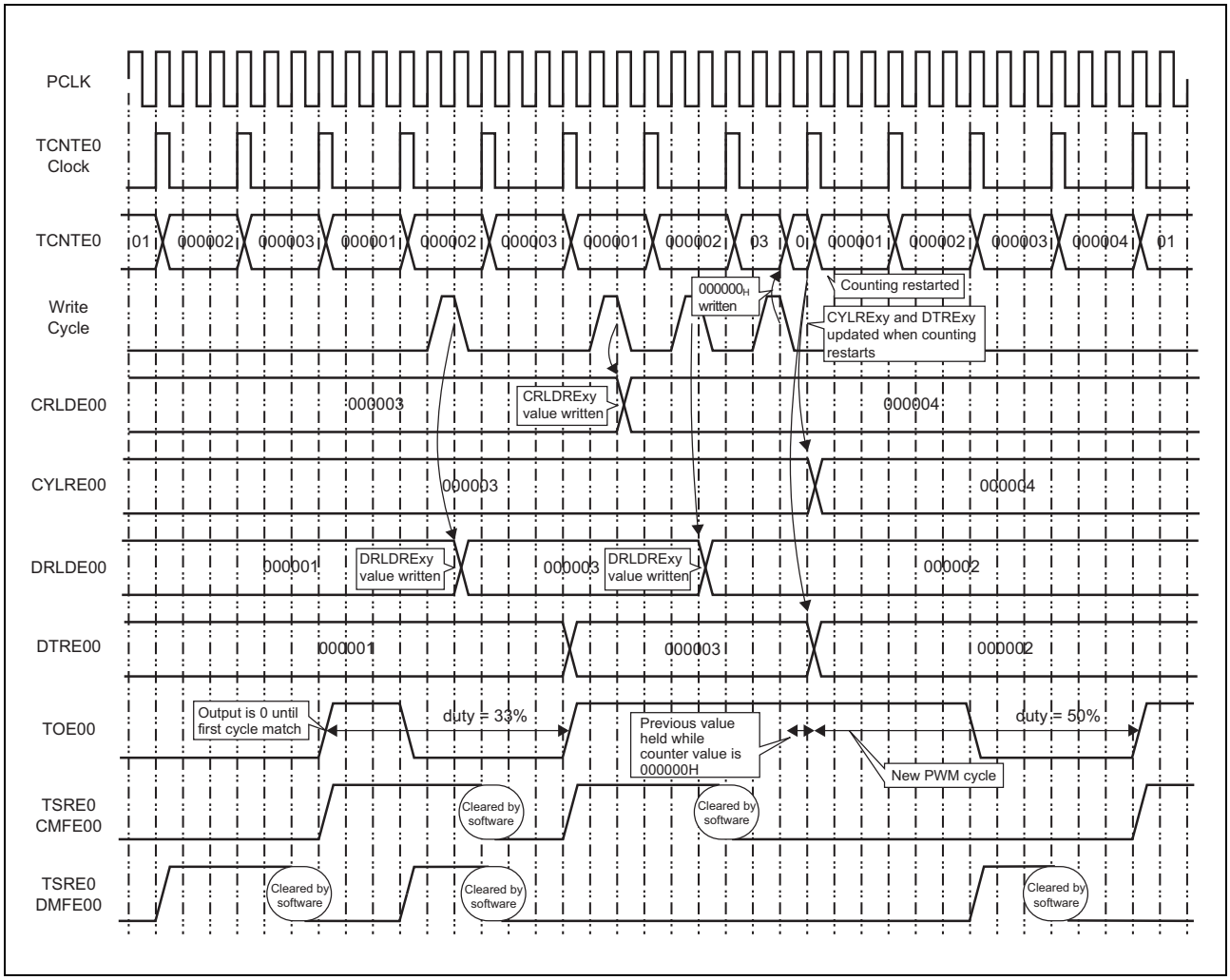


Figure 21.56 Operation of PWM (2)

Writing 000 000_H to the 24 higher-order bits of the counter changes the 24 higher-order bits values of the counter to 000 000_H. The output waveform (TOE00) does not change. If the reload feature is enabled, an input of counter clock after writing 000 000_H to the 24 higher-order bits transfers the values of the cycle reload register and the duty reload register to the cycle-setting register and the duty cycle setting register. At the same time, counting starts and PWM output also starts.

A waveform is output in off-state duty (active-low output) mode by selecting the off-state duty mode in timer output control register Ex (TOCREx). The output waveform on pin TOExy is inverted on the next PCLK cycle after setting.

Figure 21.57 shows an example of a waveform when switching on- and off-state duty modes. By selecting the off-state duty mode before the counter is started, the initial output level on the PWM output pin TOE00 is 1. After the counter started until the first cycle match, the level on pin TOE00 retains 1. On the following cycle match and duty cycle match, the output levels are alternated.

When the PWM cycle is forcibly terminated by writing 000 000_H to the 24 higher-order bits of the counter, TOE00 retains the previous value. At the timing in which the 24 higher-order bits of the counter is incremented to 000 001_H, a new PWM cycle is started.

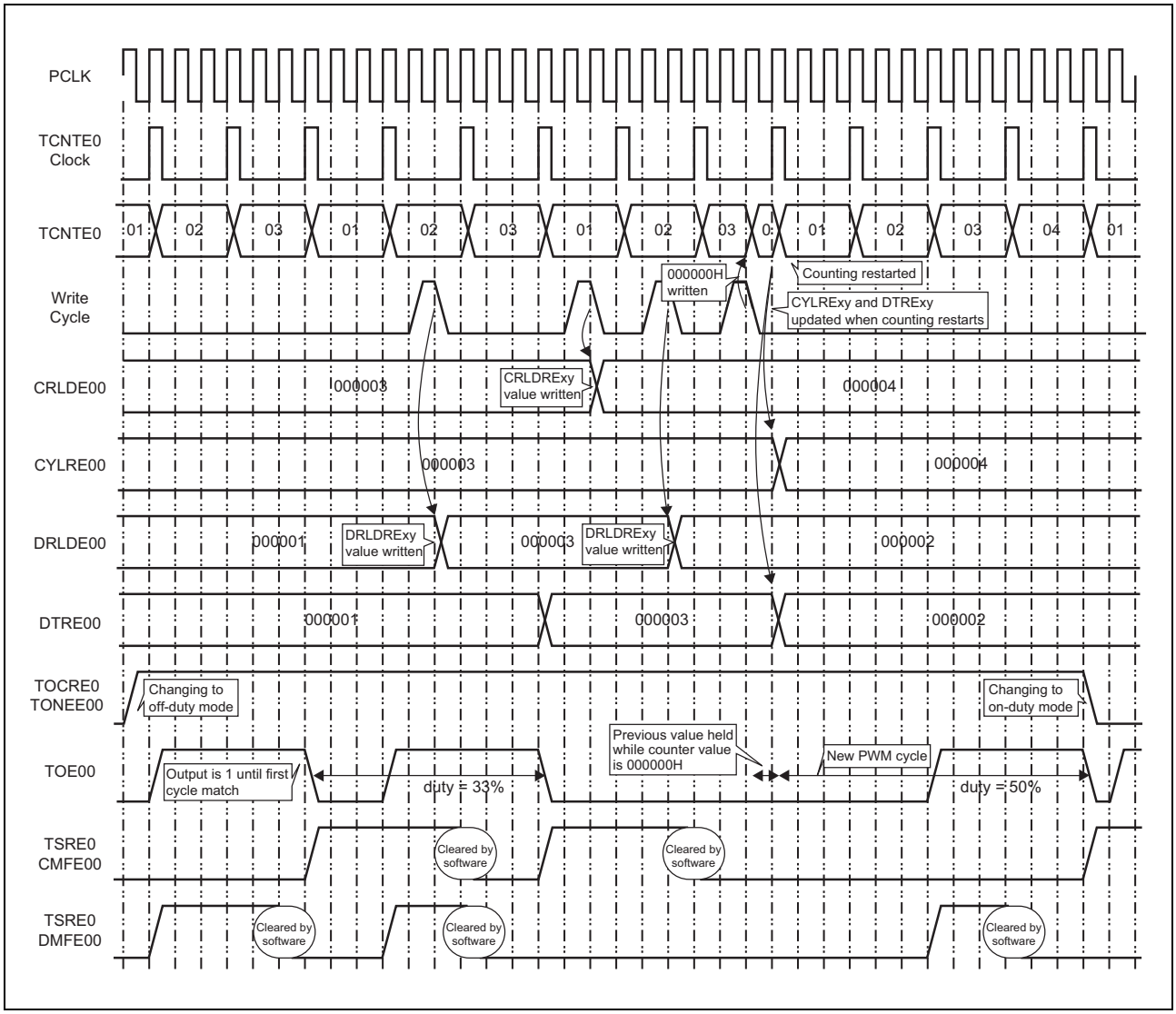


Figure 21.57 Operation of PWM (3)

Figure 21.58 illustrates the output waveform when PWM output is shutoff. PWM output is shutoff if the output shutoff enable/disable selection bit (POEENxy) of the output-shutoff control register Ex (POECREx) is enabled and shutoff input (POEx) is enabled during a duty period (between a cycle match in the previous PWM cycle to a duty match). POEx being enabled during non-duty period does not trigger a shutoff. During an output shutoff period, the signal is output to TOExy at the level specified by the output level H/L selection bit (PWMSLVxy) of the output shutoff setting register Ex (SOLVLEx). This output level is not affected by the inversion setting (on-state/off-state duty) of the corresponding bit of the timer output control register Ex (TOCREx).

The state of output shutoff is released if POEx is disabled at the time of a cycle match. The state of PWM output being shutoff continues to the next cycle match if POEx is in the enabled state at the above-mentioned timing.

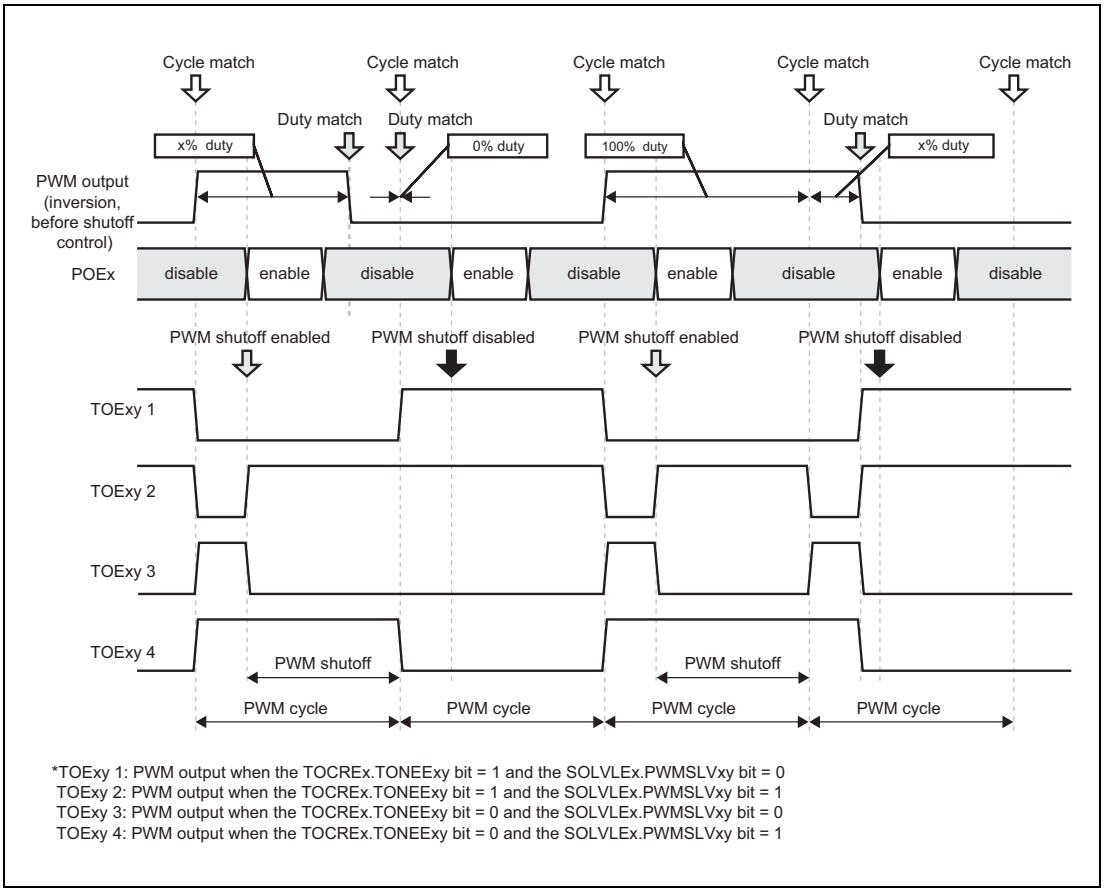


Figure 21.58 Example of PWM Output shutoff

21.9 Timer F

21.9.1 Operation Overview

The timer F block consists of 16 timer F subblocks.

Timer F subblocks realize the following features:

- Edge counting in a specified period:
  Counts the number of edges input to the external input pin (TIFxA or TIFx)
- Effective edge interval counting:
  Measures time until a specified number of edges is input to the external pin (TIFxA or TIFx).
- Measurement of time during high/low input levels:
  Measures a total amount of time when a high or low level is input to the external input pin (TIFxA or TIFx). The duration of measurement is designated as the number of pulses input to the external pin.
- Measurement of PWM input waveform timing:
  Measures the off-duty period and cycle time of the PWM waveform input to the external pin (TIFxA or TIFx). The duration of measurement is designated as the number of PWM cycles input to the external pin.
- Rotation speed/pulse measurement (for the subblock 3 to 15 only):
  Every time an edge is input to the external pin (TIFx), the following values are retained edge count, time stamp at edge input, edge input interval (cycle), and high/low input level immediately before input.
- Up/down event count (for the subblock 0 to 2 only):
  TIFnA of the two external pins (TIFxA, TIFxB) is used to count as the count source. TIFnB switches between up-counting and down-counting.
- Four-time multiplication event count (for the subblock 0 to 2 only):
  Counting operation is executed using two external input pins (TIFxA, TIFxB) as the count sources. Signals in the pins switch between up-counting and down-counting.

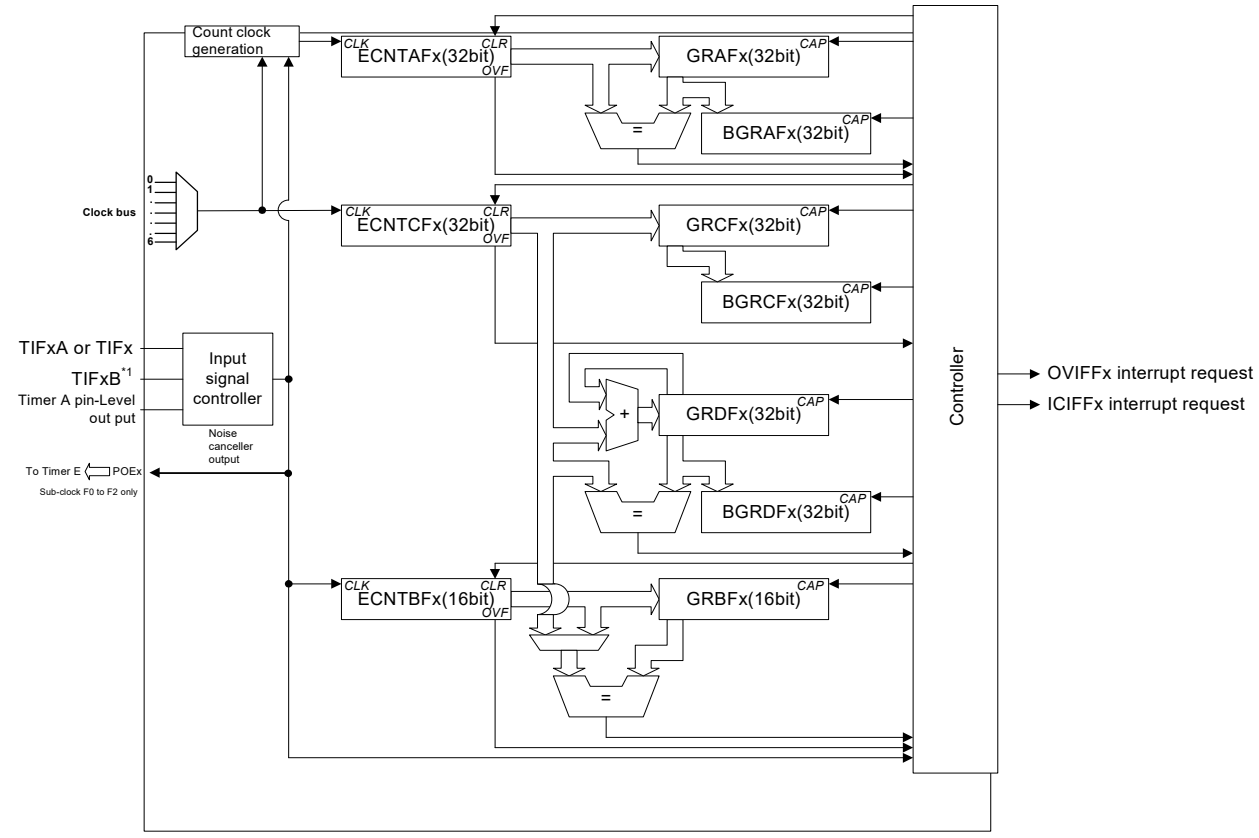
Input signals from the external input pins TIFxA or TIFx and TIFxB can be subject to the noise cancellation function using the input cancellation function. Noise-canceled signals from TIA00 to TIA06 of timer A can be used instead of external signal input pin TIFxA or TIFx.

Inputs of TIF0A to TIF2A and TIF0B and TIF2B can also be used as inputs (shutoff inputs) for POE0 to POE5 of timer E. For the correspondence between TIFxA/B and POEx, see **Table 21.96** of timer E.

Note that, if any of TIF0A to TIF2A and TIF0B to TIF2B is used as a shutoff input of timer E, the shutoff input becomes the measurement target of the corresponding timer F subblock.

Configuration

Figure 21.59 is a block diagram of timer F. Subblocks 0 to 2 of timer F have two external inputs (TIFxA and TIFxB) and the other subblocks have one external input (TIFx), two 32-bit counters (ECNTAFx and ECNTCFx), three 32-bit generic registers (GRAFx, GRCFx, and GRDFx), three 32-bit backup registers (BGRAFx, BGRCFx, and BGRDFx), a 16-bit up/down counter (ENCTBFx), a 16-bit generic register (GRBFx), an input processor (edge detection and noise canceler), and a controller.



Note 1. TIFxB is for subblocks 0 to 2 only.

Figure 21.59 Block Diagram of Subblocks of Timer F

NOTE

For the correspondence between TIFxA/B and POEx, see **Table 21.96** of timer E.

Interrupt request

The timer F can output two types of interrupts totaling 32 interrupts.

- OVFFx interrupt requests (16 interrupts ($0 \leq x \leq 15$)): An interrupt is output when one of the three counters (ECNTAFx, ECNTBFx, ECNTCFx) in the subblock Fx has overflown or underflown (only in ECNTBFx). They can also be used as compare match interrupts in PWM input waveform measurement mode and “rotation speed/pulse measurement” mode. To which counter the interrupt belongs can be known by referring to the timer status register F (ISRF). This request is received by the INTC block and the designated processing is performed.
- ICFFx interrupt requests (16 interrupts ($0 \leq x \leq 15$)): The interrupt is output when a count value capturing in the timer subblock Fx occurs. This request is received by the DMAC block or INTC. DMA transfer by DMAC enables to transfer captured data obtained by using compare match as a trigger to the on-chip RAM or perform designated processing by interrupt request. For details on DMA transfer by DMAC, see **Section 7, DMA**.

21.9.2 Registers Related to Timer F

21.9.2.1 TSTRF — Timer Start Register F

Access: 8-bit access, 16-bit access is possible.

Address: FFE6 3000_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	STRF 15	STRF 14	STRF 13	STRF 12	STRF 11	STRF 10	STRF 9	STRF 8	STRF 7	STRF 6	STRF 5	STRF 4	STRF 3	STRF 2	STRF 1	STRF 0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.109 TSTRF Register Contents

Bit Position	Bit Name	Function
15 to 0	STRF15 to STRF0	Timer F start 0: Stops counting of ECNTAFx, ECNTBFx, and ECNTCFx. 1: Permits counting of ECNTAFx, ECNTBFx, and ECNTCFx.

Note: x is an integer from 0 to 15.

TSTRF is a 16-bit readable/writable register that specifies whether to operate or stop each subblock (timer F0 to F15) in the timer F. Count operation is not executed unless TFE bit in ATUIV master enable register (ATUENR) is enabled even if the start bit in timer F is set to enable the count operation.

When TSTRF is reset, it is initialized to 0000_H.

(1) STRFx — Counter F Start

These bits specify whether to operate or stop two time counters (ECNTAFx, ECNTCFx) in subblocks (F0 to F15) of timer F and event counter Fx (ECNTBFx). When this bit is cleared to 0, ECNTAFx, ECNTBFx, and ECNTCFx stop operation. Counter value is retained at stop state. When this bit is set to 1 once again, the operation starts at the retained value.

Count operation is not executed unless TFE bit in ATU-IV master enable register (ATUENR) is enabled even if the start bit in timer F is set enable the count operation.

CAUTION

The prescaler is operating regardless of the setting of the counter F start bit, and not initialized at the start of counter. Therefore, during the time between the activation and the start of actual count operation by the above counter, hardware-related uncertainty shorter than the period of selected count source (resolution) accompanies

21.9.2.2 NCMCR1F — Noise Cancellation Mode Channel Register 1F

Access: 8-bit access and 16-bit access are possible.

Address: FFE6 3004_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	NCM1F 15	NCM1F 14	NCM1F 13	NCM1F 12	NCM1F 11	NCM1F 10	NCM1F 9	NCM1F 8	NCM1F 7	NCM1F 6	NCM1F 5	NCM1F 4	NCM1F 3	NCM1F 2	NCM1F 1	NCM1F 0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.110 NCMCR1F Register Contents

Bit Position	Bit Name	Function
15 to 0	NCM1F 15 to NCM1F 0	Noise Cancel Mode Specifies the operation mode of the noise canceler of channel x. 0: Premature-transition cancellation mode 1: Minimum time-at-level cancellation mode (when NCMF = 0 and NCM2Fx = 0) Level accumulation cancellation mode (when NCMF = 0 and NCM2Fx = 1)

Note: x stands for the subblock number (channel number) of 0 to 15.

NCMCR1F is a 16-bit readable/writable register that selects the mode of operation of the noise canceller for each channel unit.

In the premature-transition cancellation mode, after a change in the level of the input signal has been detected, further changes in the level of the input signal that occur within a specified period are ignored. That is, changes with a specified period after each initial level change are regarded as noise in this mode.

In minimum time-at-level cancellation mode, the first and subsequent level changes are ignored unless the input signal level remains the same over a given period. Level changes occurring within a shorter period are considered to indicate an unstable signal, and such signals are treated as noise.

In level accumulation cancellation mode, input signal levels are accumulated until it reaches 0 or the specified value, at which time the input level is considered to have reached to 0 or 1. The period is set by noise canceler registers in each of the applicable blocks and is counted by a noise canceler counter.

Figure 21.1 illustrates the premature-transition cancellation mode, and **Figure 21.3** illustrates overview of the noise cancellation operation (TIA00 input of timer A) in level accumulation cancellation mode. Each channel detects edges in signals that have been processed by the noise canceler. **Figure 21.1** and **Figure 21.3** illustrate the cases when rising edges are detected in signals processed by the noise canceler.

(1) NCM1Fx — Channel n noise cancellation mode

Specify the operation mode of noise canceller in channel x.

CAUTION

This register is effective only when the NCMF bit of the noise cancel mode register (NCMR) of the common controller is set to 0. The channel mode when the corresponding bit (NCM1Fx) of this register is set to 1 can be set to minimum time-at-level cancellation mode or level accumulation cancellation mode depending on the setting of the corresponding bit of the noise cancel mode channel register 2F (NCMCR2F). Note: x is from 0 to 15.

Table 21.111 Noise Cancel Mode Correspondence of Timer F

Channel Enable	Filter Mode		Channel Register			
NCCRF. NCEFx	NCMR. NCMF	NCMR. NCMSEL	NCMCR1F. NCM1Fx	NCMCR2F. NCM2Fx	Operation Mode	Filter Unit
0	—	—	—	—	Filter disabled	—
1	0	(—)	0	(—)	Premature-transition	Each subblock/ channel
1	0	(—)	1	0	Minimum time-at-level	Each channel
1	0	(—)	1	1	Level accumulation	Each channel
1	1	(0)	—	—	Minimum time-at-level	Each subblock
1	1	1	—	—	Level accumulation	Each subblock

Note 1. x is from 0 to 15.

Note 2. A noise canceller enable bit Fx (NCEFx) can be set for each channel.

Setting when timer F filtering is made for each channel (Prerequisites: channel enable = 1)

- Each channel setting: When the noise cancel mode bit (NCMF) of the noise cancel mode register (NCMR) is set to 0 and any one of the noise cancel mode setting bits (NCM1Fx) of the noise cancel mode channel register 1F (NCMCR1F) of timer F is set to 1.
- All channel setting: Same as the above description except that the noise cancel mode bit of the noise cancel mode register is set to 1. Or, in the above description, all noise cancel mode setting bits (NCM1Fx) of the noise cancel mode channel register 1F (NCMCR1F) are set to 0.

21.9.2.3 NCMCR2F — Noise Cancellation Mode Channel Register 2F

Access: 8-bit access, and 16-bit access are possible.

Address: FFE6 3008_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	NCM2F15	NCM2F14	NCM2F13	NCM2F12	NCM2F11	NCM2F10	NCM2F9	NCM2F8	NCM2F7	NCM2F6	NCM2F5	NCM2F4	NCM2F3	NCM2F2	NCM2F1	NCM2F0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.112 NCMCR2F Register Contents

Bit Position	Bit Name	Function
15 to 0	NCM2F15 to NCM2F0	Noise Cancel Mode Specifies the operation mode of the noise canceler of channel x. 0: Minimum time-at-level cancellation mode (when NCMF = 0 and NCM1Fx = 1) 1: Level accumulation cancellation mode (when NCMF = 0 and NCM1Fx = 1)

Note: x stands for the subblock number (channel number) of 0 to 15.

Noise cancel mode channel register 2F is a 16-bit readable/writable register, which selects the operation mode of the noise canceler for each channel. In level accumulation cancellation mode, input signal levels are accumulated until it reaches 0 or the specified value, at which time the input level is considered to have reached to 0 or 1.

Intervals are specified by the noise cancel register of each channel, and the time is measured by the noise cancel counter.

Figure 21.1 illustrates the premature-transition cancellation mode, and **Figure 21.3** illustrates overview of the noise cancellation operation (TIA00 input of timer A) in level accumulation cancellation mode. Each channel detects edges in signals that have been processed by the noise canceler. **Figure 21.1** and **Figure 21.3** illustrate the cases when rising edges are detected in signals processed by the noise canceler.

(1) NCM2Fx — Channel n noise cancellation mode

Specify the operation mode of noise canceller in channel x.

CAUTION

This register is effective only when the NCMF bit of the noise cancel mode register (NCMR) of the common controller is set to 0. The channel mode when the corresponding bit (NCM1Fx) of this register is set to 1 can be set to the minimum time-at-level cancellation mode or the level accumulation cancellation mode depending on the setting of the corresponding bit of the noise cancel mode channel register 2F (NCMCR2F). Note: x is from 0 to 15.

The following table indicates the true/false conditions of the noise cancel mode.

Table 21.113 Noise Cancel Mode Correspondence of Timer F

Channel Enable	Filter Mode		Channel Register			
NCCRF. NCEFx	NCMR. NCMF	NCMR. NCMSEL	NCMCR1F. NCM1Fx	NCMCR2F. NCM2Fx	Operation Mode	Filter Unit
0	—	—	—	—	Filter disabled	—
1	0	(—)	0	(—)	Premature-transition	All channels/Each channel
1	0	(—)	1	0	Minimum time-at-level	Each channel
1	0	(—)	1	1	Level accumulation	Each channel
1	1	(0)	—	—	Minimum time-at-level	All channels
1	1	1	—	—	Level accumulation	All channels

Note 1. x is from 0 to 15.

Note 2. A noise canceler enable bit Fx(NCEFx) can be set for each channel.

Setting when timer F filtering is made for each channel (Prerequisites: channel enable = 1)

- Each channel setting: When the noise cancel mode bit (NCMF) of the noise cancel mode register (NCMR) is set to 0 and any one of the noise cancel mode setting bits (NCM1Fx) of the noise cancel mode channel register 1F (NCMCR1F) of timer F is set to 1.
- All channel setting: Same as the above description except that the noise cancel mode bit of the noise cancel mode register is set to 1. Or, in the above description, all noise cancel mode setting bits (NCM1Fx) of the noise cancel mode channel register 1F (NCMCR1F) are set to 0.

21.9.2.4 NCCRF — Noise Canceller Control Register F

Access: 8-bit access and 16-bit access are possible.

Address: FFE6 300C$_H$

Value after reset: 0000$_H$

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	NCEF 15	NCEF 14	NCEF 13	NCEF 12	NCEF 11	NCEF 10	NCEF 9	NCEF 8	NCEF 7	NCEF 6	NCEF 5	NCEF 4	NCEF 3	NCEF 2	NCEF 1	NCEF 0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.114 NCCRF Register Contents

Bit Position	Bit Name	Function
15 to 0	NCEFx	Noise Canceler Enable Fx (NCEFx) 0: The noise cancel feature of TIFxA or TIFx and TIFxB is disabled. 1: The noise cancel feature of TIFxA or TIFx and TIFxB is enabled.

Note: x is an integer from 0 to 15.

The noise canceler control register F (NCCRF) is a 16-bit readable/writable register that specifies enable/disable of the noise canceler feature of each subblock (timer F0 to timer F15) of timer F.

When NCCRF is reset, it is initialized to 0000$_H$.

(1) NCEFx — Noise Canceller Enable Fx

Specify to enable/disable the noise canceller in each subblock. Regarding the subblocks 2 to 0, each subblock has noise cancellers TIFxA and TIFxB but enabling/disabling these cancellers cannot be specified independently. Setting the NCEFx bit to 1 enables each noise canceller in TIFxA and TIFxB.

If the noise cancel feature is enabled, upon detection of a level change of the input signal of external input TIFxA, TIFx, or TIFxB, the settings in the noise cancel mode register (NCMR) of common controller, noise cancel mode channel register 1F (NCMCR1F) of timer F, and nose cancel mode channel register 2F (NCMCR2F) of timer F start operation by selecting an operation mode from premature-transition cancellation mode, minimum time-at-level cancellation mode, and level accumulation cancellation mode.

In premature-transition cancellation mode, when the input signal level change is detected, the change is output as the signal that has passed through noise canceling.

Simultaneously, corresponding noise canceler counters FAx, FBx(NCNTFA15 to NCNTFA0 and NCNTFB2 to NCNTFB0) start up-counting. The input signal level change is masked until a compare match occurs between the value in the noise canceler counter and the values in the noise cancel register (NCRFA15 to NCRFA0 and NCRFB2 to NCRFB0). When a compare match occurs, the input signal level at this moment is output as the signal after noise canceling.

When these bits are cleared to 0 while NCNTFAx and NCNTFBx are in count operation, the count operation continues until a compare match occurs and the level change of the corresponding external input (TIFxA or TIFx and TIFxB) is kept being masked.

In minimum time-at-level cancellation mode, when the level change of the input signal is detected, the corresponding noise canceler counter FAx, FBx (NCNTFA15 to NCNTFA 0 and NCNTFB2 to NCNTFB0) starts up-counting. If a level change of input signal is not detected during the period until a compare match occurs between the value in the noise canceler counter and the values in the noise cancel register (NCRFA15 to NCRFA0 and NCRFB2 to NCRFB0), the level change at the compare match is output as the signal after a noise cancellation. If a noise change is detected, this change is regarded as noise and the noise canceller does not change the signal after a noise cancel regarding that no level change of input signal occurred.

When these bits are cleared to 0 while NCNTFAx and NCNTFBx are in count operation, the count operation continues to keep noise canceling processing until a compare match or input signal change occurs.

In level accumulation cancellation mode, the input signal level determines the up/down counting of the corresponding noise cancel counters FAx, FBx (NCNTFA0 to NCNTFA15 and NCNTFB0 to NCNTFB2). Up-counting is made with high-level input, down-counting is made with low-level inputs. Up-count continues until a compare match between the noise cancel counter and the noise cancel registers (NCRFA0 to NCRFA15 and NCRFB0 to NCRFB2), and down-count continues until the noise cancel counter matches $*0000_H$. If an up-counting results in a compare match (NCRFA0 to NCRFA15 and NCRFB0 to NCRFB2), the noise canceler output is updated to 1. If a down-counting results in a compare match (0000_H), the noise canceler output is updated to 0.

In minimum time-at-level cancellation mode and premature-transition cancellation mode, level change detection is always made by Pϕ regardless of the selected noise cancel clock. In level accumulation cancellation mode, the sampling of the input level is made by the noise canceler clock (Pϕ or Pϕ /128).

For an operating example in cancel mode, see **Figure 21.1**, **Figure 21.2** and **Figure 21.3**.

21.9.2.5 PVFCRF — Private Function Control Register F

Access: 16-bit access is possible.

Address: FFE6 3010_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PVFCRKEY[7:0]								—	—	—	—	—	—	GRDFCMEN	BKCRWEN
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R/W*1	R	R	R	R	R	R	R/W	R/W

Note 1. Values written to the write key code are not retained.

Table 21.115 PVFCRF Register Contents

Bit Position	Bit Name	Function
15 to 8	PVFCRKEY [7:0]	Write Key Code $C9_H$: GRDFCMEN and BKCRWEN values can be changed. Other than $C9_H$: GRDFCMEN and BKCRWEN values cannot be changed.
7 to 2	—	Not used. These bits are always read as 0. Fix these bits to 0.
1	GRDFCMEN	GRDFx Compare Match Enable 0: GRDFx compare match feature is disabled in PWM input waveform measurement mode. 1: GRDFx compare match feature is enabled in PWM input waveform measurement mode.
0	BKCRWEN	Backup Control Register Fx Write Enable 0: The BKCRFx register is not writable. 1: The BKCRFx register is writable.

Note: x is an integer from 0 to 15.

The private function control register F (PVFCRF) is a 16-bit readable/writable register. To write a value to this register, write key code setting is required to bits 15 to 8 as well as the data to be written.

When PVFCRF is reset, it is initialized to 0000_H.

(1) PVFCRKEY — Write Key Code

This code controls the write permission for bits GRDFCMEN and BKCRWEN. Setting is required to bits 15 to 8 as well as the data to be written to bits GRDFCMEN and BKCRWEN. Data written to this bit is not retained; Value 00_H is always returned when being read.

(2) GRDFCMEN — GRDFx Compare Match Enable

Controls enable/disable of the compare match of time measurement counter CFx (ECNTCFx) and general register DFx (GRDFx). This bit takes effect in PWM input waveform mode, and is ignored in other modes. (GRDFx compare match feature is disabled.)

The setting of this bit is applied to all subblocks of timer F. Subblock-individual setting is not possible.

(3) BKCRWEN — Backup Control Register Fx Write Enable

Controls write permission to the backup control register Fx (BKCRFx) The setting of this bit is applied to all subblocks of timer F. Subblock-individual setting is not possible.

21.9.2.6 TCR1Fx — Timer Control Registers 1Fx

(x = 0 to 15: Correspond to subblocks F0 to F15.)

Access: 8-bit access is possible.

Address: FFE6 3040_H + (40_H * x)

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	CKSELFx[2:0]			MDFx[2:0]			EGSELFx[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.116 TCR1Fx Register Contents

Bit Position	Bit Name	Function
7 to 5	CKSELFx[2:0]	Clock Select Fx Specify the clock sources for the time counters (ECNTAFx, ECNTCFx). 000: Clock bus 0 001: Clock bus 1 010: Clock bus 2 011: Clock bus 3 100: Clock bus 4 101: Clock bus 5 110: Clock bus 6 111: Reserved
4 to 2	MDFx[2:0]	Timer Operation Mode Fx Specify the operation mode for the corresponding subblocks Fx. 000: Edge counting in a specified period 001: Effective edge interval counting 010: Measurement of time during high/low input levels 011: Reserved 100: Measurement of PWM input waveform timing 101: Rotation speed/pulse measurement 110: Up/down event count 111: Four-time multiplication event count
1, 0	EGSELFx[1:0]	Edge Select Fx Specify the edge sense mode of TIFxA, TIFx, or TIA00-06 input. 00: Edge detection disabled 01: Rising edge 10: Falling edge 11: Both edges

Timer control registers 1Fx (TCR1F0 to TCR1F15) are 8-bit readable/writable registers that specify the operation mode of the subblocks F0 to F15.

When TCR1F0 to TCR1F15 are reset, they are initialized to 00_H.

(1) CKSELFx — Clock Select Fx

Specify the clock sources for the two time counters (ECNTAFn, ECNTCFn) in the subblocks F0 to F15.

Setting these bits to a value from 000_B to 110_B selects the corresponding clock bus (0 to 6) as the clock source. Do not specify 111_B. If specified, the operation is not guaranteed.

(2) MDFx — Timer Operation Mode Fx

Specify the operation mode for the timer subblocks F0 to F15. There are seven modes: up/down event count, four-time multiplication event count, edge counting in a specified period, effective edge interval counting, measurement of time during high/low input levels, measurement of PWM input waveform timing, and rotation speed/pulse measurement.

Do not set rotation speed/pulse measurement for subblocks other than 3 to 15.

(3) EGSELFx — Edge Select Fx

Specifies the event input (TIFxA or TIFx) of timer subblock F0 to F15 and the edge sense mode of pin inputs (TIA00 to 06) from timer A. Edge detection is done for signals that have passed through the noise canceller. Therefore, edge detection is done to the external input (TIFxA or TIFx, TIFxB) if the noise cancel function is disabled, and to signals after noise cancel if the noise cancel function is enabled. Noise cancellation for TIA00 to TIA06 input signals is in accord with the setting of noise cancellation for timer A.

While measurement of time during high/low input levels is specified, when this bit selects the falling edge, measurement of time during high level is specified. When this bit selects the rising edge, measurement of time during low level is specified. Do not select both edges.

While measurement of PWM input waveform timing and rotation speed/pulse measurement are specified, when this bit selects the rising edge, the period between the two rising edges is regarded as the PWM cycle and the low level period is regarded as the off-duty period. If the falling edge is selected, the period between the two falling edges is regarded as the PWM cycle and the high-level period is regarded as the off-duty period. Do not select both edges.

When up/down event count mode and four-time multiplication event count mode are specified, be sure to designate both the rising and falling edges. If otherwise selected, the operation is not guaranteed.

CAUTION

TIFxB pin is available only when up/down event count and four-time multiplication event count are specified. TIFxB operates always detecting both the rising and falling edges. In other modes, TIFxB does not detect edges.

21.9.2.7 TCR2Fx — Timer Control Registers 2Fx

(x = 0 to 15: Corresponding to subblocks F0 to F15)

Access: 8-bit access is possible.

Address: FFE6 3042_H + (40_H * x)

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	EISELEFx	—	—	—	—	EISELFx		
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R	R/W	R/W	R/W

Table 21.117 TCR2Fx Register Contents

Bit Position	Bit Name	Function
7 to 5	EISELEFx	Event Input Selection Enable Ex 0: Selects TIFxA or TIFx. 1: Selects the signal processed by noise canceling of TIA.
6 to 3	—	Not used. Fix these bits to 0.
2 to 0	EISELFx	Event Input Selection Ex 000: Selects the signal of TIA00 of timer A processed by noise canceling. 001: Selects the signal of TIA01 of timer A processed by noise canceling. 010: Selects the signal of TIA02 of timer A processed by noise canceling. 011: Selects the signal of TIA03 of timer A processed by noise canceling. 100: Selects the signal of TIA04 of timer A processed by noise canceling. 101: Selects the signal of TIA05 of timer A processed by noise canceling. 110: Selects the signal of TIA06 of timer A processed by noise canceling. 111: Setting not permitted.

The timer control register 2Fx (TCR2Fx) is an 8-bit readable/writable register that specifies the event input of each subblock (timer F0 to timer F15) of timer F.

When TCR2F0 to TCR2F15 are reset, they are initialized to 00_H.

(1) EISELEFx — Event Input Selection Enable Fx

Selects an event input for each subblock (timer F0 to timer F15) included in timer F.

By setting this bit, noise-canceled signals from TIA00 to TIA06 of timer A can be used as the event input instead of external input pin TIFxA or TIFx.

(2) EISELFx — Event Input Selection Fx

When EISELEFx is set to 1, noise-canceled signal TIA00 to TIA06 of timer A can be selected as the event input for each subblock (timer F0 to timer F15) included in timer F.

Do not specify 111_B. If specified, the operation is not guaranteed.

21.9.2.8 TIERFx — Timer Interrupt Enable Registers Fx

(x = 0 to 15: Corresponding to subblocks F0 to F15)

Access: 8-bit access is possible.

Address: FFE6 3047_H + (40_H * x)

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	OVECFx	OVEBFx	OVEAFx	—
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R

Table 21.118 TIERFx Register Contents

Bit Position	Bit Name	Function
7 to 4	—	Not used. These bits are always read as 0. When writing, always write 0.
3	OVECFx	Overflow Interrupt Enable CFx 0: Interrupt by OVFCFx disabled 1: Interrupt by OVFCFx enabled
2	OVEBFx	Overflow Interrupt Enable BFx 0: Interrupt by OVFBFx disabled 1: Interrupt by OVFBFx enabled
1	OVEAFx	Overflow Interrupt Enable AFx 0: Interrupt by OVFAFx disabled 1: Interrupt by OVFAFx enabled
0	—	Not used. This bit is always read as 0. These bits are always read as 0. Fix this bit to 0.

Timer interrupt enable registers Fx (TIERF0 to TIERF15) are 8-bit readable/writable registers that specify whether to enable or disable the interrupt corresponding to the status flag of the timer status register Fx (TSRFx).

When TIERF0 to TIERF15 are reset, they are initialized to 00_H.

(1) Overflow Interrupt Enable AFx

Specifies whether to enable or disable the interrupt by OVFAFx to the status corresponding to the overflow of the time measurement counter AFx (ECNTAFx).

(2) Overflow Interrupt Enable BFx

Specifies whether to enable or disable the interrupt by OVFBFx to the status corresponding to the overflow/underflow of the event counter Fx (ECNTBFx).

(3) Overflow Interrupt Enable CFx

This flag specifies whether to permit or prohibit interrupt requests of the state (OVFCFx) corresponding to overflow of timer measurement counter CFx (ECNTCFx) (when it is in PWM input waveform measurement mode and the GRDFCMEN bit of the PVFCRF register is 0), corresponding to compare match of ECNTCFx and GRDFx (when it is in PWM input waveform measurement mode and the GRDFCMEN bit of the PVFCRF register is 1), or corresponding to compare match of ECNTCFx and GRBFx (rotation speed/pulse measurement mode).

CAUTION

The overflow of interrupt of the timer subblock Fx is requested as the logical sum of the interrupts OVFAFx, OVFBFx, and OVFCFx. By referring to TSRFx, which counter generated the interrupt by overflow or underflow can be known. Clear the corresponding interrupt request flag of TSRFx when processing the interrupt. The interrupt request will be repeated if the flag is not cleared

21.9.2.9 BKCRFx — Backup Control Register Fx

(x = 0 to 15: Corresponding to subblocks F0 to F15)

Access: 8-bit access is possible.

Address: FFE6 3044_H + (40_H * x)

Value after reset: 00_H

[Subblocks F00 to F02]

Bit	7	6	5	4	3	2	1	0
	—	—	BKENCFx	—	—	—	ARSWCFx	—
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R	R	R	R/W	R

[Subblocks F03 to F15]

Bit	7	6	5	4	3	2	1	0
	—	BKENAFx	BKENCFx	BKENDFx	—	ARSWAFx	ARSWCFx	ARSWDFx
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R/W*2	R/W*2	R/W*2	R	R/W*2	R/W*2	R/W*2

Table 21.119 BKCRFx Register Contents

Bit Position	Bit Name	Function
7	—	Not used. This bit is always read as 0. When writing, always write 0.
6	BKENAFx*1	Backup Enable A Saving GRAFx in BGRAFx when CDRFx is read 0: Not saved 1: Saved
5	BKENCFx	Backup Enable C Saving GRCFx in BGRCFx when CDRFx is read 0: Not saved 1: Saved
4	BKENDFx*1	Backup Enable D Saving GRDFx in BGRDFx when CDRFx is read 0: Not saved 1: Saved
3	—	Not used. This bit is always read as 0. When writing, always write 0.
2	ARSWAFx*1	Mapping to Access Register Switching A To FFE6 3054_H + (40_H * x), 0: GRAFx is mapped. 1: BGRAFx is mapped.
1	ARSWCFx	Mapping to Access Register Switching C To FFE6 3064_H + (40_H * x), 0: GRCFx is mapped. 1: BGRCFx is mapped.
0	ARSWDFx*1	Mapping to Access Register Switching D To FFE6 3064_H + (40_H * x), 0: GRDFx is mapped. 1: BGRDFx is mapped.

Note 1. BKENAFx, BKENDFx, ARSWAFx, and ARSWDFx exist only in subblocks F03 to F15. Nothing is deployed in F00 to F02. These bits are always read as 0. Writing value must always be 0.

Note 2. Writing to this register is permitted only when the BKCRWEN bit of the PVFCRF register is 1.

The backup control register Fx (BKCRF00 to BKCRF15) is an 8-bit readable/writable register.

Writing to BKCRF00 to BKCRF15 is permitted only when the BKCRWEN bit of the private function control register F (PVFCRF) is 1.

When BKCRF00 to BKCRF15 are reset, they are initialized to 00_H.

(1) BKENAFx, BKENCFx, and BKENDFx — Backup Enable A, C, and D

These bits permit or prohibit value saving of GRAFx to BGRAFx, GRCFx to BGRCFx, and GRDFx to BGRDFx when capture output register Fx (CDRFx) is read.

(2) ARSWAFx, ARSWCFx, and ARSWDFx — Access Register Switching A, C, and D

These bits specify whether BGRAFx is mapped instead of GRAFx to address FFE6 $3054_H + (40_H * x)$, whether BGRCFx is mapped instead of GRCFx to address FFE6 $3064_H + (40_H * x)$, and whether BGRDFx is mapped instead of GRDFx to address FFE6 $3068_H + (40_H * x)$.

CAUTION

Setting of this register always takes effect regardless of the operation mode. Value saving and mapping change are performed even for registers that do not operate in some operation mode.

21.9.2.10 TSRFx — Timer Status Registers Fx

(x = 0 to 15: Corresponding to subblocks F0 to F15)

Access: 8-bit access is possible.

Address: FFE6 3045_H + (40_H * x)

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	OVFCFx	OVFBFx	OVFAFx	ICFFx
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 21.120 TSRFx Register Contents

Bit Position	Bit Name	Function
7 to 4	—	Not used. These bits are always read as 0. When writing, always write 0.
3	OVFCFx	Overflow/Compare Match Flag CFx 0: No overflow in ECNTCFx 1: ECNTCFx overflows
2	OVFBFx	Overflow Flag BFx 0: No overflow or underflow in ECNTBFx 1: ECNTBFx overflows or underflows
1	OVFAFx	Overflow Flag AFx 0: No overflow in ECNTAFx 1: ECNTAFx overflows
0	ICFFx	Input Capture Flag Fx 0: Input capture is not detected in the subblock Fx 1: Input capture in subblock Fx detected

Timer status registers Fx (TSRF0 to TSRF15) are 8-bit readable/writable registers that indicate overflows in the time counters A and C, overflow or underflow in the event counter, and input capture occurrence.

These flags are status flags that indicate occurrence of an interrupt request. For overflow flags AFx, BFx, and CFx (OVFAx, OVFBx, and OVFCx), interrupt requests can be issued by setting the TIERFx register. Setting a bit of the timer status clear register Fx (TSCRFx) clears the corresponding flag.

For input capture flag Fx (ICFFx), an interrupt request is made again If an interrupt source takes effect while this flag is set. An interrupt request is issued even in cases of contention between clearing by the corresponding timer status clear register and setting by an interrupt source.

For overflow flags, an interrupt request is issued when the corresponding flag is set by setting the corresponding bit of timer interrupt enable register Fx (TIERFx). To cancel an interrupt request, clear this flag by the timer status clear register Fx (TSCRFx).

When TSRF00 to TSRF15 are reset, they are initialized to 00_H.

(1) ICFFx — Input Capture Flag Fx

By this bit, the detection state of input capture and compare match in the subblock Fx (F0 to F15) can be monitored. This flag cannot be set to 1 or 0 by software.

- Setting condition
  When input capture is detected in the subblock Fx.
- Clearing condition
  When 1 is written to ICFCFx of the timer status clear register Fx (TSCRFx).

(2) OVFAFx — Overflow Flag AFx

By this flag, overflow of the time counter AFx (ECNTAFx) can be monitored. This flag cannot be set to 1 or 0 by software.

- Setting condition
  When ECNTAFx overflows (FFFF $FFFF_H \rightarrow 0000\ 0000_H$)
- Clearing condition
  When 1 is written to OVFCAFx of the timer status clear register Fx (TSCRFx).

(3) OVFBFx — Overflow Flag BFx

By this flag, overflow or underflow of the event counter Fx (ECNTBFx) can be monitored. This flag cannot be set to 1 or 0 by software.

- Setting condition
  When ECNTBFx overflows ($FFFF_H \rightarrow 0000_H$) or underflows ($0000_H \rightarrow FFFF_H$)
- Clearing condition
  When 1 is written to OVFCBFx of the timer status clear register Fx (TSCRFx).

(4) OVFCFx — Overflow Flag CFx

Values of this flag indicate different conditions depending on the operation mode and the setting of the GRDFx compare match enable bit (GRDFCMEN) of the private function control register F (PVFCRF). When the PWM input waveform measurement mode is set, value 0 of the GRDFCMEN bit makes this flag indicate an overflow of time measurement counter CFx (ECNTCFx), and value 1 of the GRDFCMEN bit makes this flag indicate a compare match of ENCTCFx and GRDFx. In “rotation speed/pulse measurement” mode, this flag indicates a compare match of ECNTCFx and GRBFx regardless of the GRDFCMEN bit setting.

This flag cannot be set to 1 or 0 by software.

- Setting condition
  Measurement of PWM input waveform timing mode
  [GRDFCMEN = 0]
  When the value of ECNTCFx has overflown (from FFFF $FFFF_H$ to 0000 0000_H)
  [GRDFCMEN = 1]
  When the values of ECNTCFx and GRDFx coincide.
  Rotation speed/pulse measurement mode
  When ECNTCFx and GRDFx (the value of zero extension to the 16 lower-order bits) coincide
- Clearing condition
  When 1 is written to OVFCCFx of the timer status clear register Fx (TSCRFx).

21.9.2.11 TSCRFx — Timer Status Clear Register Fx

(x = 0 to 15: Corresponding to subblocks F0 to F15)

Access: 8-bit access is possible.
Value 0 cannot be written. The value is not retained when 1 is written. These bits are always read as 0.

Address: FFE6 3046_H + (40_H * x)

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	OVFCCFx	OVFCBFx	OVFCAFx	ICFCFx
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 21.121 TSCRFx Register Contents

Bit Position	Bit Name	Function
7 to 4	—	Not used. Fix these bits to 0.
3	OVFCCFx	Overflow Flag Clear CFx 0: Disabled (Initial value) 1: Clears OVFCFx of timer status register Fx (TSRFx) to 0.
2	OVFCBFx	Overflow Flag Clear BFx 0: Disabled (Initial value) 1: Clears OVFBFx of timer status register Fx (TSRFx) to 0.
1	OVFCAFx	Overflow Flag Clear AFx 0: Disabled (Initial value) 1: Clears OVFAFx of timer status register Fx (TSRFx) to 0.
0	ICFCFx	Input Capture Flag Clear Fx 0: Disabled (Initial value) 1: Clears ICFFx of timer status register Fx (TSRFx) to 0.

The timer status clear register Fx (TSCRF00 to TSCRF15) is an 8-bit readable/writable register. This register specifies flag clearing in the events of an overflow of time measurement counter A or C, an overflow or an underflow of an event counter, and an input capture occurrence.

TSCRFx is readable/writable only in 8-bit units. This register is always read as 0.

When TSCRFx is reset, it is initialized to 00_H.

(1) OVFCCFx — Overflow Flag Clear CFx

When overflow flag CFx (OVFCFx) of timer status register Fx (TSRFx) is set to 1, writing 1 to this register clears OVFCFx to 0. This register is always read as 0.

(2) OVFCBFx — Overflow Flag Clear BFx

When overflow flag BFx (OVFBFx) of timer status register Fx (TSRFx) is set to 1, writing 1 to this register clears OVFBFx to 0. This register is always read as 0.

(3) OVFCAFx — Overflow Flag Clear AFx

When overflow flag AFx (OVFAFx) of timer status register Fx (TSRFx) is set to 1, writing 1 to this register clears OVFAFx to 0. This register is always read as 0.

(4) ICFCFx — Input Capture Flag Clear Fx

When input capture flag Fx (ICFFx) of timer status register Fx (TSRFx) is set to 1, writing 1 to this register clears ICFFx to 0. This register is always read as 0.

21.9.2.12 ECNTAFx — Timer Measurement Counters AFx

(x = 0 to 15: Corresponding to subblocks F0 to F15)

Access: 32-bit access is supported. 8-bit access and 16-bit access are prohibited.

Address: FFE6 3050_H + (40_H * x)

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ECNTAFx															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ECNTAFx															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.122 ECNTAFx Register Contents

Bit Position	Bit Name	Function
31 to 0	ECNTAFx	Time Measurement Count AFx Up-counter A

ECNTAF0 to ECNTAF15 are 32-bit readable/writable registers. This register is readable/writable only in 32-bit units. Do not write or read 8-bit or 16-bit data for this register.

This register, with one provided to each subblock, executes up-count operation using the input clock. One clock bus from clock buses 0 to 6 can be selected as the input clock according to the setting of the corresponding control register. The input clocks for ECNTAFx and ECNTCFx are the same. Clock source cannot be set independently.

When clearing the counter is done at the countup timing, ECNTAFx is cleared to 0000 0001_H, and to 0000 0000_H in other cases.

When ECNTAF0 to ECNTAF15 are reset, they are initialized to 0000 0000_H.

21.9.2.13 ECNTBFx — Event Counters Fx

(x = 0 to 15: Corresponding to subblocks F0 to F15)

Access: 16-bit access is supported. 8-bit access is prohibited.

Address: FFE6 3058_H + (40_H * x)

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ECNTBFx															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.123 ECNTBFx Register Contents

Bit Position	Bit Name	Function
15 to 0	ECNTBFx	Event Count Fx Up/down counter

Event counters Fx (ECNTBF0 to ECNTBF15) are 16-bit readable/writable registers. This register, with one provided to each subblock, executes up-count/downcount operation using the input clock. The input clock is given two external input pins (TIFxA, TIFxB) for subblocks 0 to 2 and one external input TIFx for the other subblocks. The external pin and edge used to count differs according to the setting of the corresponding control register (operation mode and edge select).

The input clock in each mode is listed in **Table 21.124**.

When clearing the counter is done at the count-up timing, ECNTBFx is cleared to 0001_H, and to 0000_H in other cases.

When ECNTBF0 to ECNTBF15 are reset, they are initialized to 0000_H.

Table 21.124 Event Counter Input Clocks and Count Edges for Each Operation Mode of Timer

Operation Mode	Input Clock	Count Edge
Edge counting in a specified period	TIFxA or TIFx	Selectable by EGSELFx
Effective edge interval counting	TIFxA or TIFx	Selectable by EGSELFx
Measurement of time during high/ low input levels	TIFxA or TIFx	Selectable by EGSELFx (other than both edges)
Measurement of PWM input waveform timing	TIFxA or TIFx	Selectable by EGSELFx (other than both edges)
Rotation speed/pulse measurement	TIFx	Selectable by EGSELFx (other than both edges)
Up/down event count	TIFxA (Count direction is specified by TIFnB level)	Both rising/falling edges
Four-time multiplication event count	TIFxA, TIFxB	Both rising/falling edges

21.9.2.14 ECNTCFx — Time measurement Counters CFx

(x = 0 to 15: Corresponding to subblocks F0 to F15)

Access: 32-bit access is supported. 8-bit access and 16-bit access are prohibited.

Address: FFE6 3060_H + (40_H * x)

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ECNTCFx															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ECNTCFx															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.125 ECNTCFx Register Contents

Bit Position	Bit Name	Function
31 to 0	ECNTCFx	Time Counter CFx Up-counter C

Timer counters CFx (ECNTCF00 to ECNTCF15) are 32-bit readable/writable registers. They are up-counters. This register is readable/writable only in 32-bit units. Do not write or read 8-bit or 16-bit data for this register.

This register, with one provided to each subblock, is enabled only in measurement of PWM input waveform timing and rotation speed/pulse measurement modes. This register does not execute count operation in other modes.

This register executes up-count operation using the input clock. One clock bus from clock buses 0 to 6 can be selected as the input clock according to the setting of the corresponding control register. The input clocks for ECNTAFx and ECNTCFx are the same. Clock source cannot be set independently.

The timing of external input and the count value clearing by a compare match with ECNTBFx are synchronized with the count clock of ECNTCFx. At this time, ECNTCFx is cleared to 0000 0001_H.

When ECNTCF0 to ECNTCF15 are reset, they are initialized to 0000 0000_H.

21.9.2.15 GRAFx — General Registers AFx

(x = 0 to 15: Corresponding to subblocks F0 to F15)

Access: 32-bit access is supported. 8-bit access and 16-bit access are prohibited.

Address: FFE6 3054_H + (40_H * x)

Value after reset: FFFF $FFFF_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GRAFx															
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GRAFx															
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.126 GRAFx Register Contents

Bit Position	Bit Name	Function
31 to 0	GRAFx	General Registers AFx Input capture value or output compare match value for the time counter A.

General registers AFx (GRAF00 to GRAF15) are 32-bit readable/writable registers. This register is readable/writable only in 32-bit units. Do not write or read 8-bit or 16-bit data for this register. This register, with one provided to each subblock, has two functions such as input capture register and output compare register for the time measurement counter AFx (ECNTAFx).

Do not set GRAFx to $0000\ 0000_H$ to function this register as the compare match register. Note that if $0000\ 0000_H$ is set, incorrect measurement may occur.

When GRAF00 to GRAF15 are reset, they are initialized to FFFF $FFFF_H$.

GRAF03 to GRAF15 are mapped to target addresses and become readable and writable when the ARSWAFx bit of backup control register Fx (BKCRFx) is 0 (initial value: 0).

21.9.2.16 BGRAFx — Backup Register AFx

(x = 3 to 15: Correspond to subblocks F3 to F15.)

Access: 32-bit access is supported. 8-bit access and 16-bit access are prohibited.

Address: FFE6 3054$_H$ + (40$_H$ * x)

Value after reset: FFFF FFFF$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BGRAFx[31:16]															
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BGRAFx[15:0]															
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 21.127 BGRAFx Register Contents

Bit Position	Bit Name	Function
31 to 0	BGRAFx[31:0]	When the BKENAFx bit of BKCRFx is 1, this register retains the value of general register AFx (GRAFx) at the time of reading the capture output register Fx (CDRFx).

Backup register AFx (BGRAF03 to BGRAF15) is a 32-bit read-only register. One instance of this register is provided for each of subblock 03 to subblock 15. If the BKENAFx bit of backup control register Fx (BKCRFx) is 1, this register retains the content of general register AFx (GRAFx) when capture output register Fx (CDRFx) is read.

When BGRAF03 to BGRAF15 are reset, they are initialized to FFFF FFFF$_H$.

BGRAFx is mapped to target addresses and becomes readable when the ARSWAFx bit of backup control register Fx (BKCRFx) is 1 (initial value: 0).

21.9.2.17 GRBFx — General Registers BFx

(x = 0 to 15: Correspond to subblocks F0 to F15.)

Access: 16-bit access is possible. 8-bit access is prohibited.

Address: FFE6 305C$_H$ + (40$_H$ * x)

Value after reset: FFFF$_H$

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GRBFx															
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.128 GRBFx Register Contents

Bit Position	Bit Name	Function
15 to 0	GRBFx	General Registers BFx Input capture value or output compare match value for event counter.

General registers BFx (GRBF00 to GRAB15) are 16-bit readable/writable registers. This register, with one provided to each subblock, has two functions such as input capture register and output compare register for the event counter Fx (ECNTBFx).

Do not set GRBFn to 0000$_H$ to function this register as the compare match register. Note that if 0000$_H$ is set, incorrect measurement may occur.

When GRBF00 to GRBF15 are reset, they are initialized to FFFF$_H$.

21.9.2.18 GRCFx — General Registers CFx

(x = 0 to 15: Corresponding to subblocks F0 to F15)

Access: 32-bit access is possible. 8-bit access and 16-bit access are prohibited.

Address: FFE6 3064_H + (40_H * x)

Value after reset: FFFF $FFFF_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GRCFx															
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GRCFx															
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.129 GRCFx Register Contents

Bit Position	Bit Name	Function
31 to 0	GRCFx	General Registers CFx Input capture value for the time counter C

General register CFx (GRCF00 to GRCF15) is a 32-bit readable/writable register, and can be read from and written to in 32-bit units. Do not write or read 8-bit or 16-bit data for this register.

This register, with one provided to each subblock, has a function as the input capture register for the time measurement counter CFx (ECNTCFx). Triggered by a compare match between ECNTBFx and GRBx (in measurement of PWM input waveform timing mode) or edge input of the TIFx pin (in rotation speed/pulse measurement mode), ECNTCFx count number is taken in at the next ECNTCFx up-count timing. These registers are valid only in measurement of PWM input waveform timing or rotation speed/pulse measurement mode. Capture operation is not executed in other modes.

When GRCF00 to GRCF15 are reset, they are initialized to FFFF $FFFF_H$.

GRCF00 to GRCF15 are mapped to target addresses and become readable and writable when the ARSWCFx bit of backup control register Fx (BKCRFx) is 0 (initial value: 0).

21.9.2.19 BGRCFx — Backup Register CFx

(x = 0 to 15: Corresponding to subblocks F0 to F15)

Access: 32-bit access is supported. 8-bit access and 16-bit access are prohibited.

Address: FFE6 3064_H + (40_H * x)

Value after reset: FFFF $FFFF_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BGRCFx[31:16]															
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BGRCFx[15:0]															
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 21.130 BGRCFx Register Contents

Bit Position	Bit Name	Function
31 to 0	BGRCFx[31:0]	When the BKENCFx bit of BKCRFx is 1, this register retains the value of general register CFx (GRCFx) at the time of reading the capture output register Fx (CDRFx).

Backup register CFx (BGRCF00 to BGRCF15) is a 32-bit read-only register.

One instance of this register is provided for each subblock. If the BKENCFx bit of backup control register Fx (BKCRFx) is 1, this register retains the content of general register CFx (GRCFx) when capture output register Fx (CDRFx) is read.

When BGRCF00 to BGRCF15 are reset, they are initialized to FFFF $FFFF_H$.

BGRCFx is mapped to target addresses and becomes readable when the ARSWCFx bit of backup control register Fx (BKCRFx) is 1 (initial value: 0).

21.9.2.20 GRDFx — General Registers DFx

(x = 0 to 15: Corresponding to subblocks F0 to F15)

Access: 32-bit access is supported. 8-bit access and 16-bit access are prohibited.

Address: FFE6 3068_H + (40_H * x)

Value after reset: FFFF $FFFF_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GRDFx															
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GRDFx															
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.131 GRDFx Register Contents

Bit Position	Bit Name	Function
31 to 0	GRDFx	General Register DFx In PWM input waveform measurement mode: Compare match value of time measurement counter C In “rotation speed/pulse measurement” mode: (only for x = 3 to 15) accumulated value of time measurement counter C

General registers DFx (GRDF00 to GRDF15) are 32-bit readable/writable registers that can be read and written only in 32-bit units.

Do not read or write these registers in 16 bit or 8-bit units.

This register is provided individually for subblocks F00 to 15. This register operates differently depending on the subblock. For subblocks F00 to F02, this register works only in PWM input waveform measurement mode. This register does not work in other modes.

For subblocks F03 to F15, this register works in PWM input waveform measurement mode and in “rotation speed/pulse measurement” mode. This register does not work in other modes.

In PWM input waveform measurement mode, this register works as a compare match register, and is always compared with the value of time measurement counter CFx (ECNTCFx). If the values of both registers coincide, the OVFCFx bit of timer status register Fx (TSRFx) is set to 1 at the next Pϕ clock cycle. To make this register work as a compare match register, the GRDFCMEN bit of PVFCRF must be set. In rotation speed/pulse measurement mode, this register works as an input capture register. Triggered by an edge input of the TIFx pin, this register takes and accumulates the value of time measurement counter Cx (ECNTCFx) at the next cycle of ECNTCFx increment.

When GRDF00 to GRDF15 are reset, they are initialized to FFFF $FFFF_H$.

GRDF03 to GRDF15 are mapped to target addresses and become readable and writable when the ARSWDFx bit of backup control register Fx (BKCRFx) is 0 (initial value: 0).

21.9.2.21 BGRDFx — Backup Register DFx

(x = 3 to 15: Corresponding to subblocks F3 to F15)

Access: 32-bit access is supported. 8-bit access and 16-bit access are prohibited.

Address: FFE6 3068_H + (40_H * x)

Value after reset: FFFF $FFFF_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BGRDFx[31:16]															
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BGRDFx[15:0]															
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 21.132 BGRDFx Register Contents

Bit Position	Bit Name	Function
31 to 0	BGRDFx[31:0]	When the BKENDFx bit of BKCRFx is 1, this register retains the value of general register DFx (GRDFx) at the time of reading the capture output register Fx (CDRFx).

Backup register DFx (BGRDF03 to BGRDF15) is a 32-bit read-only register. This register is readable only in 32-bit units. Do not read this register in 8-bit or 16-bit units.

One instance of this register is provided for each of subblock F03 to subblock F15. If the BKENDFx bit of backup control register Fx (BKCRFx) is 1, this register retains the content of general register DFx (GRDFx) when capture output register Fx (CDRFx) is read.

When BGRDF03 to BGRDF15 are reset, they are initialized to FFFF $FFFF_H$.

BGRDFxis mapped to target addresses and becomes readable when the ARSWDFx bit of backup control register Fx (BKCRFx) is 1 (initial value: 0).

21.9.2.22 CDRFx — Capture Output Registers Fx

(x = 0 to 15: Corresponding to subblocks F0 to F15)

Access: 32-bit access is possible.
8-bit access and 16-bit access are prohibited.

Address: FFE6 306C$_H$ + (40$_H$ * x)

Value after reset*1: 0000 FFFF$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CDRFx															
Value after reset*1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CDRFx															
Value after reset*1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Note 1. The value after a reset is 0000 FFFF$_H$ since the timer operation mode after a reset is “edge counting in a specified period” (i.e., the MDFx bit in the TCR1Fx register = 000$_B$).

Table 21.133 CDRFx Register Contents

Bit Position	Bit Name	Function
31 to 0	CDRFx	Capture Output Register Fx Data retained in GRAFx, GRBFx, or ECNTBFx is read depending on the operation mode. Edge count mode: GRBFx Edge input interval measurement mode: GRAFx Input high period measurement mode: GRAFx Input low period measurement mode: GRAFx PWM input waveform measurement mode: GRAFx Rotation speed/pulse measurement mode: ECNTBFx Up/down count mode: GRBFx Four-time multiplication event count mode: GRBFx

The capture output register is a 32-bit read-only register. This register is provided to each subblock. When this register is read, values in GRAFx, GRBFx, or ECNTBFx is read according to the operation mode. A 16-bit value in GRBFx and ECNTBFx are read from the 16 lower-order bits in CDRFx. In this case, the 16 higher-order bits in CDRFx are read as 0.

Registers corresponding to various modes are listed below. Writing to these registers are ignored.

Edge count mode: GRBFx

Edge input interval measurement mode: GRAFx

Input high period measurement mode: GRAFx

Input low period measurement mode: GRAFx

PWM input waveform measurement mode: GRAFx

Rotation speed/pulse measurement mode: ECNTBFx

Up/down count mode: GRBFx

Four-time multiplication event count mode: GRBFx

When CDRF0 to CDRF15 are reset, they are initialized to FFFF FFFF$_H$.

Depending on the setting of bits BKENAFx, BKENCFx, and BKENDFx of backup control register Fx (BKCRFx), reading CDRFx triggers value saving from general register AFx (GRAFx) to backup general register AFx (BGRAFx), from general register CFx (GRCFx) to backup general register CFx (BGRCFx), and from general register DFx (GRDFx) to backup general register DFx (BGRDFx). For more information, see the sections on backup general registers corresponding to individual general registers.

21.9.2.23 NCNTFAx — Noise Canceler Counters FAx

(x = 0 to 15: Corresponding to subblocks F0 to F15)

Access: 16-bit access is supported. 8-bit access is prohibited.

Address: FFE6 3048_H + (40_H * x)

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	NCNTFAx															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.134 NCNTFAx Register Contents

Bit Position	Bit Name	Function
15 to 0	NCNTFAx	Noise Cancel Counter FAx 16-bit count value

Noise canceler counters registers FAx (NCNTFAx) are 16-bit readable/writable registers.

When the noise canceler is enabled by noise canceler control register F (NCCRF) and the operation mode is premature-transition cancellation mode or minimum time-at-level cancellation mode, a level change of external input pin TIFxA or TIFx triggers the up-count operation ticked by the noise canceler counter clock provided by the prescaler. In level accumulation cancellation mode, up/down counting is performed based on the external input level.

NCNTFAx is readable/writable only in 16-bit units.

When NCNTFAx is reset, it is initialized to 0000_H. One of the three different operation modes, which are premature-transition cancellation mode, minimum time-at-level cancellation mode, and level accumulation cancellation mode, takes place depending on the setting of the noise cancel mode selection bit (NCMSEL) and the timer F noise cancel mode bit (NCMF) of the noise cancel mode register (NCMR) of the common controller, as well as on a couple of timer F bits, which are the noise cancel mode bit (NCM1Fx) of the noise cancel mode channel register 1F (NCMCR1F) and the noise cancel mode bit (NCM2Fx) of the noise cancel mode channel register 2F (NCMCR2F).

- Premature-transition cancellation mode
  NCNTFAx starts up-count operation triggered by the level change of input signal in TIFxA or TIFx under the condition that the NCEFx bit is set to 1 and NCNTFAx is not in count operation. When the count number matches the value in the noise cancel register FAx (NCRFAx), this register stops the count operation, clearing the count value to 0000_H synchronizing with the next PCLK. NCNTFAx executes the count operation regardless of the setting of the TFE bit in the ATU-IV master enable register (ATUENR).
  A level change at the start of count operation is output as it is as the signal that passed through the noise canceling operation. Although this signal is the subject of edge detection, the signal does not change because any input level change is masked until the count number matches the value of NCRFAx.
  Even if the NCEFx bit is cleared during the count operation, the count operation continues until the count number matches the value of NCRFAx. The input signal is masked during all that time.

- Minimum time-at-level cancellation mode
  NCNTFAx starts up-count operation triggered by the level change of input signal in TIFxA or TIFx under the condition that the NCEFx bit is set to 1 and NCNTFAx is not in count operation.
  When the level change of the input signal occurs during the count operation or the count number matches the value in the noise cancel register FAx (NCRFAx), this register stops the count operation, clearing the count value to 0000_H synchronizing with the next PCLK.
  NCNTFAx executes the count operation regardless of the setting of the TFE bit in the ATU-IV master enable register (ATUENR).
  Signals that passed through the noise canceling operation can change only when the count number matches the value of NCRFAx according to the level change at the count start. If the count operation stops before the count number matches the value of NCRFAx, signals that passed through the noise canceling operation do not change because the level changes at the start or stop of counting are masked.
  Even if the NCEFx bit is cleared during the count operation, the count operation and noise canceling processing continue until a compare match occurs or the level of the input signal changes.

- Level accumulation cancellation mode
  When the NCEFx bit is set to 1, NCNTFAx performs up or down counting depending on the input signal level. Up-counting is performed when the input level is high, and if the counter value coincides with NCRFAx, the up-counting stops at the next PCLK cycle. Down-counting is performed when the input level is low, and if the counter value coincides with 0000_H, the down-counting stops at the next PCLK cycle.
  Counting of NCNTFAx is performed regardless of the setting of the TFE bit of the ATU-IV master enable register (ATUENR).
  If up-counting results in a compare match of NCRFAx, the noise canceler output is updated to 1. If down-counting results in a compare match with 0000_H, the noise canceler output is updated to 0.
  If the NCEFx bit is cleared when counting is in progress, the noise cancel counter stops counting, and the value is changed from the noise canceler output to the current input signal level. Be noted that clearing the NCEFx bit in level accumulation cancellation mode might cause an edge detection due to this value change.

21.9.2.24 NCNTFBx — Noise Canceler Counters FBx

(x = 0 to 2: Correspond to subblocks F0 to F2.)

Access: 16-bit access is supported. 8-bit access is prohibited.

Address: FFE6 304C$_H$ + (40$_H$ * x)

Value after reset: 0000$_H$

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	NCNTFBx															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.135 NCNTFBx Register Contents

Bit Position	Bit Name	Function
15 to 0	NCNTFBx	Noise Cancel Counter FBx 16-bit count value

Noise cancel counters FBx (NCNTFBx) are 16-bit readable/writable registers. These registers are available only in up/down event count mode and 4-time multiplication event count mode.

When the noise canceler is enabled by noise canceler control register F (NCCRF) and the operation mode is premature-transition cancellation mode or minimum time-at-level cancellation mode, a level change of external input pin TIFxB triggers the up-count operation ticked by the noise canceler counter clock provided by the prescaler. In level accumulation cancellation mode, up/down counting is performed based on the external input level.

NCNTFBx is readable/writable only in 16-bit units.

When NCNTFBx is reset, it is initialized to 0000$_H$.

One of the three different operation modes, which are premature-transition cancellation mode, minimum time-at-level cancellation mode, and level accumulation cancellation mode, takes place depending on the setting of the noise cancel mode selection bit (NCMSEL) and the timer F noise cancel mode bit (NCMF) of the noise cancel mode register (NCMR) of the common controller, as well as on a couple of timer F bits, which are the noise cancel mode bit (NCM1Fx) of the noise cancel mode channel register 1F (NCMCR1F) and the noise cancel mode bit (NCM2Fx) of the noise cancel mode channel register 2F (NCMCR2F).

- Premature-transition cancellation mode
  NCNTFBx starts upcount operation triggered by the level change of input signal in TIFxB under the condition that the NCEFx bit is set to 1 and NCNTFBx is not in count operation. When the count number matches the value in the noise cancel register FBx (NCRFBx), this register stops the count operation, clearing the count value to 0000$_H$ synchronizing with the next PCLK. NCNTFBx executes the count operation regardless of the setting of the TFE bit in the ATU-IV master enable register (ATUENR).
  A level change at the start of count operation is output as it is as the signal that passed through the noise canceling operation. Although this signal is the subject of edge detection, the signal does not change because any input level change is masked until the count number matches the value of NCRFBx.
  Even if the NCEFx bit is cleared during the count operation, the count operation continues until the count number matches the value of NCRFBx. The input signal is masked during all that time.

- Minimum time-at-level cancellation mode
  NCNTFBx starts upcount operation triggered by the level change of input signal in TIFxB under the condition that the NCEFx bit is set to 1 and NCNTFBx is not in count operation. When the level change of the input signal occurs during the count operation or the count number matches the value in the noise cancel register FBx (NCRFBx), this register stops the count operation, clearing the count value to 0000_H synchronizing with the next PCLK.
  NCNTFBx executes the count operation regardless of the setting of the TFE bit in the ATU-IV master enable register (ATUENR).
  Signals that passed through the noise canceling operation can change only when the count number matches the value of NCRFBx, according to the level change at the count start. If the count operation stops before the count number matches the value of NCRFBx, signals that passed through the noise canceling operation do not change because the level changes at the start or stop of counting are masked.
  Even if the NCEFx bit is cleared during the count operation, the count operation and noise canceling processing continue until a compare match occurs or the level of the input signal changes.

- Level accumulation cancellation mode
  When the NCEFx bit is set to 1, NCNTFBx performs up or down counting depending on the input signal level. Up-counting is performed when the input level is high, and if the counter value coincides with NCRFBx, the up-counting stops at the next PCLK cycle. Down-counting is performed when the input level is low, and if the counter value coincides with 0000_H, the down-counting stops at the next PCLK cycle.
  Counting of NCNTFBx is performed regardless of the setting of the TFE bit of the ATU-IV master enable register (ATUENR).
  If an up-count results in a compare match of NCRFBx, the noise canceler output is updated to 1. If down-counting results in a compare match with 0000_H, the noise canceler output is updated to 0.
  If the NCEFx bit is cleared when counting is in progress, the noise cancel counter stops counting, and the value is changed from the noise canceler output to the current input signal level.
  Be noted that clearing the NCEFx bit in level accumulation cancellation mode might cause edge detection due to this value change.

21.9.2.25 NCRFAx — Noise Cancel Registers FAx

(x = 0 to 15: Corresponding to subblocks F0 to F15)

Access: 16-bit access is supported. 8-bit access is prohibited.

Address: FFE6 304A$_H$ + (40$_H$ * x)

Value after reset: 0000$_H$

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	NCTFAx															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.136 NCRFAx Register Contents

Bit Position	Bit Name	Function
15 to 0	NCTFAx	Noise Cancel Time FAx TIFxA or TIFx noise cancel period (16-bit compare value)

Noise cancel registers FAx (NCRFAx) are 16-bit readable/writable registers that set the upper limit of the noise canceler counter FAx (NCNTFAx). When a period of 128 PCLK cycles is selected as the noise cancel clock, setting of FFFF$_H$ can cancel noise for maximum of 0.21 second period (when PCLK = 40 MHz).

One of the three different operation modes, which are premature-transition cancellation mode, minimum time-at-level cancellation mode, and level accumulation cancellation mode, takes place depending on the setting of the noise cancel mode selection bit (NCMSEL) and the timer F noise cancel mode bit (NCMF) of the noise cancel mode register (NCMR) of the common controller, as well as on a couple of timer F bits, which are the noise cancel mode bit (NCM1Fx) of the noise cancel mode channel register 1F (NCMCR1F) and the noise cancel mode bit (NCM2Fx) of the noise cancel mode channel register 2F (NCMCR2F).

- Premature-transition cancellation mode
  While NCNTFAx is in count operation, the level change of the subsequent input signal is masked. Values in NCNTFAx and NCRFAx are always compared. If a compare match occurs, these registers clear the value in NCNTFAx synchronizing with the next PCLK, stop the count operation, and cancel the masking of the input signal.
- Minimum time-at-level cancellation mode
  While NCNTFAx is in count operation, noise canceler processing waiting state is entered. Values in NCNTFAx and NCRFAx are always compared. If a compare match occurs, these registers clear the value in NCNTFAx synchronizing with the next PCLK, stop the count operation, and then cancel the masking of the input signal and the noise canceler outputs the input signal that has passed through the noise canceling processing.
- Level accumulation cancellation mode
  When NCNTFAx up-counting is in progress, values of CNTFAx and NCRFAx are compared. When a compare match occurs, the up-counting of NCNTFAx stops at the next PCLK cycle. When NCNTFAx down-counting is in progress, NCNTFAx is compared with 0000$_H$.

NCRFAx is readable/writable only in 16-bit units.

When NCRFAx is reset, it is initialized to 0000$_H$.

21.9.2.26 NCRFBx — Noise Cancel Registers FBx

(x = 0 to 2: Correspond to subblocks F0 to F2.)

Access: 16-bit access is supported. 8-bit access is prohibited.

Address: FFE6 304E$_H$ + (40$_H$ * x)

Value after reset: 0000$_H$

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	NCTFBx															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.137 NCRFBx Register Contents

Bit Position	Bit Name	Function
15 to 0	NCTFBx	Noise Cancel Time FBx TIFxB noise cancel period (16-bit compare value)

Noise cancel registers FBx (NCRFBx) are 16-bit readable/writable registers that set the upper limit of the noise canceler counter FBx (NCNTFBx). When a period of 128 PCLK cycles is selected as the noise cancel clock, setting of FFFF$_H$ can cancel noise for maximum of 0.21 second period (when PCLK = 40 MHz). This register is effective only during up/down counting and the four-time multiplication event count mode.

One of the three different operation modes, which are premature-transition cancellation mode, minimum time-at-level cancellation mode, and level accumulation cancellation mode, takes place depending on the setting of the noise cancel mode selection bit (NCMSEL) and the timer F noise cancel mode bit (NCMF) of the noise cancel mode register (NCMR) of the common controller, as well as on a couple of timer F bits, which are the noise cancel mode bit (NCM1Fx) of the noise cancel mode channel register 1F (NCMCR1F) and the noise cancel mode bit (NCM2Fx) of the noise cancel mode channel register 2F (NCMCR2F).

- Premature-transition cancellation mode
  While NCNTFBx is in count operation, the level change of the subsequent input signal is masked. Values in NCNTFBx and NCRFBx are always compared. If a compare match occurs, these registers clear the value in NCNTFBx synchronizing with the next PCLK, stop the count operation, and cancel the masking of the input signal.
- Minimum time-at-level cancellation mode
  While NCNTFBx is in count operation, noise canceler processing waiting state is entered. Values in NCNTFBx and NCRFBx are always compared. If a compare match occurs, these registers clear the value in NCNTFBx synchronizing with the next PCLK, stop the count operation. Simultaneously the noise canceler outputs the input signal that has passed through noise canceling processing.
- Level accumulation cancellation mode
  When NCNTFBx up-counting is in progress, values of CNTFBx and NCRFBx are compared. When a compare match occurs, the up-counting of NCNTFBx stops at the next PCLK cycle. When NCNTFBx down-counting is in progress, NCNTFBx is compared with 0000$_H$.

NCRFBx is readable/writable only in 16-bit units.

When NCRFBx is reset, it is initialized to 0000$_H$.

21.9.3 Detailed Operation Description

21.9.3.1 Edge Counting in a Given Time

When a period over which edges are counted is set in GRAFx, the number of edges within the period is obtained in GRBFx. When no edge is detected within the period, 0 is set to GRBFx. The period set to count is equivalent to the cycle of the ECNTAFx clock (GRAFx value). Operation of the timer Fx is described below. **Figure 21.60** shows an operation example. In this example, eight edges are input to 12 cycles of the count source clock. Timer counter ECNTAFx and event counter ECNTBFx are driven by the ECNTAFx and ECNTBFx clocks, respectively.

- ECNTAFx: Measures time using one of the clock buses 0 to 6. When a compare match is detected, the count value is cleared synchronized with the next PCLK.
- ECNTBFx: Counts edges of the signals provided from TIFxA or TIFx input. Edge types subject to count can be selected from among rising, falling, or both edges. The example given here counts the falling edges. A delay of two cycles in TIFxA or TIFx occurs because of synchronization processing. When a compare match in ECNTAFx is detected, the count number is cleared synchronized with the next PCLK. In a case where edges subject to count are given simultaneously at a count clearing by a compare match, both operations are regarded to be done in one cycle, setting the count value to 0001_H. **Figure 21.61** shows an example of this.
- GRAFx: Functions as the compare match register for ECNTAFx. A compare match is detected when the count values in ECNTAFx and GRAFx agree.
- GRBFx: Functions as the capture register for ECNTBFx. When a compare match in ECNTAFx is detected, this register captures the ECNTBFx count number synchronizing with the next PCLK.
- Compare match interrupt request output: After compare match detection of ECNTAFx, a compare match interrupt request is issued to CPU at the next PCLK cycle.
- DMA transfer request output: After compare match detection of ECNTAFx, a DMA transfer request is issued to DMAC at the next PCLK cycle.
- ICFFn flag: After detecting a compare match in ECNTAFx, sets the ICFFx flag synchronized with the next PCLK.
- ECNTCFx, GRCFx, GRDFx: Do not function

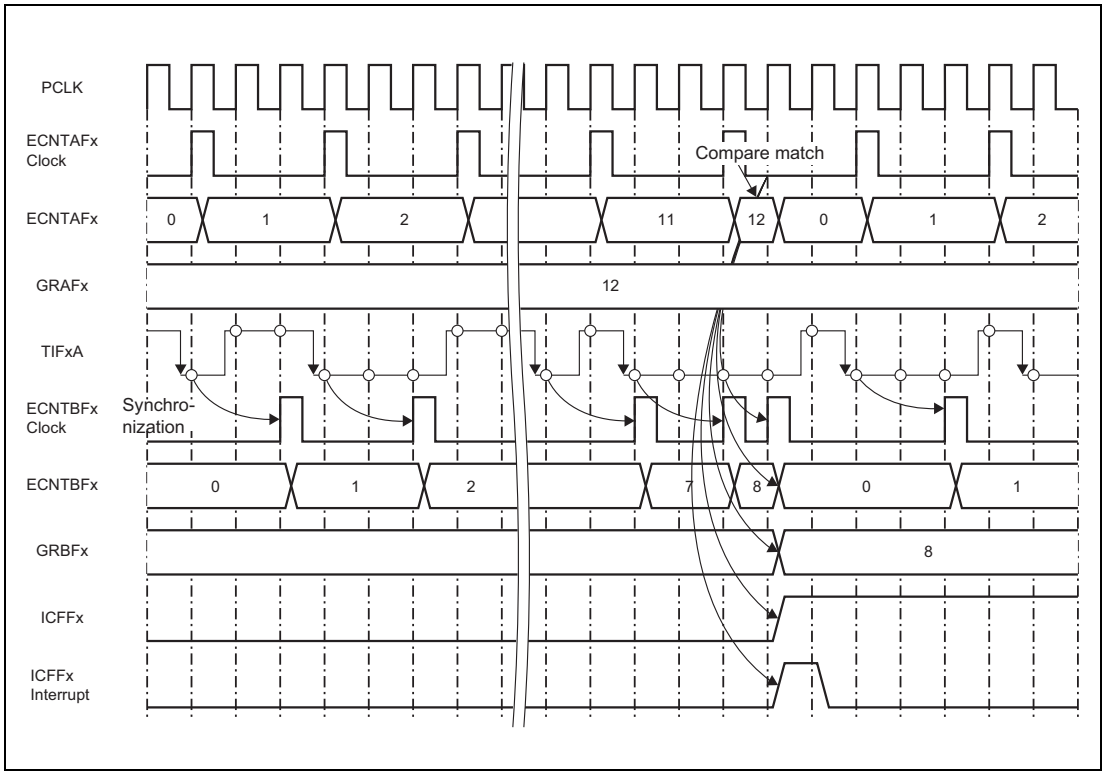


Figure 21.60 Operation Example of Edge Count in a Given Time

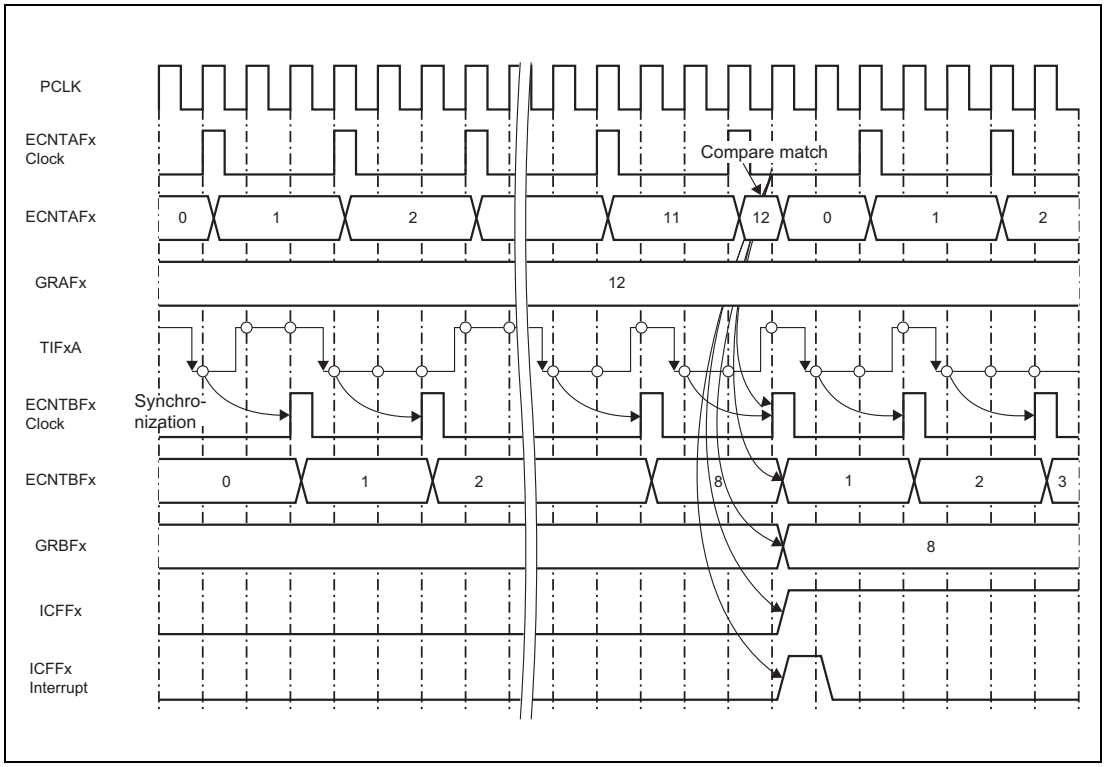


Figure 21.61 Operation Example 2 of Edge Count in a Given Time (Same Timing as Compare Match and Event)

21.9.3.2 Effective Edge Interval Counting

When a number of edges are set in GRBFx, the time necessary to count these edges is notified to GRAFx. The average of input edge intervals is obtained by dividing the time by the number of edges. The outcome is given as the unit of the ECNTAFx count source clock (GRAFx). Operation of the timer Fx is described below. **Figure 21.62** shows an operation example. In this example, 13 cycles of the counter clock are needed to detect 12 input edges. Timer counter ECNTAFx and event counter ECNTBFx are driven by the ECNTAFx and ECNTBFx clocks, respectively.

- ECNTAFx: Measures time using one of the clock buses 0 to 6. When a compare match between ECNTBFx and GRBFx is detected, the count value is cleared synchronized with the next ECNTAFx clock. Since ECNTAFx count clear occurs at the same time with countup, the cleared value becomes 0000 0001_H.
- ECNTBFx: Counts edges provided from TIFxA or TIFx. Edge types subject to count can be selected from among rising, falling, or both edges. The example given here counts the falling edges. A delay of two cycles in TIFxA or TIFx occurs because of synchronization processing. When a compare match is detected, the count number is cleared synchronized with the next PCLK.
- GRAFx: Functions as the capture register for ECNTAFx. When a compare match in ECNTBFx is detected, this register captures the ECNTAFx count number synchronizing with the next ECNTAFx clock.
- GRBFx: Functions as the compare match register for ECNTBFx. A compare match is detected when the count values in ECNTBFx and GRBFx agree.
- Compare match interrupt request: After a compare match detection of ECNTBFx, a compare match interrupt request is issued to CPU at the next “ECNTAFx Clock” cycle.
- DMA transfer request output: After a compare match detection of ECNTBFx, a DMA transfer request is issued to DMAC at the next “ECNTAFx Clock” cycle.
- ICFFx: After detecting a compare match in ECNTBFx, sets the ICFFx flag synchronized with the next ECNTAFx clock.
- ECNTCFx GRCFx GRDFx: Do not function.

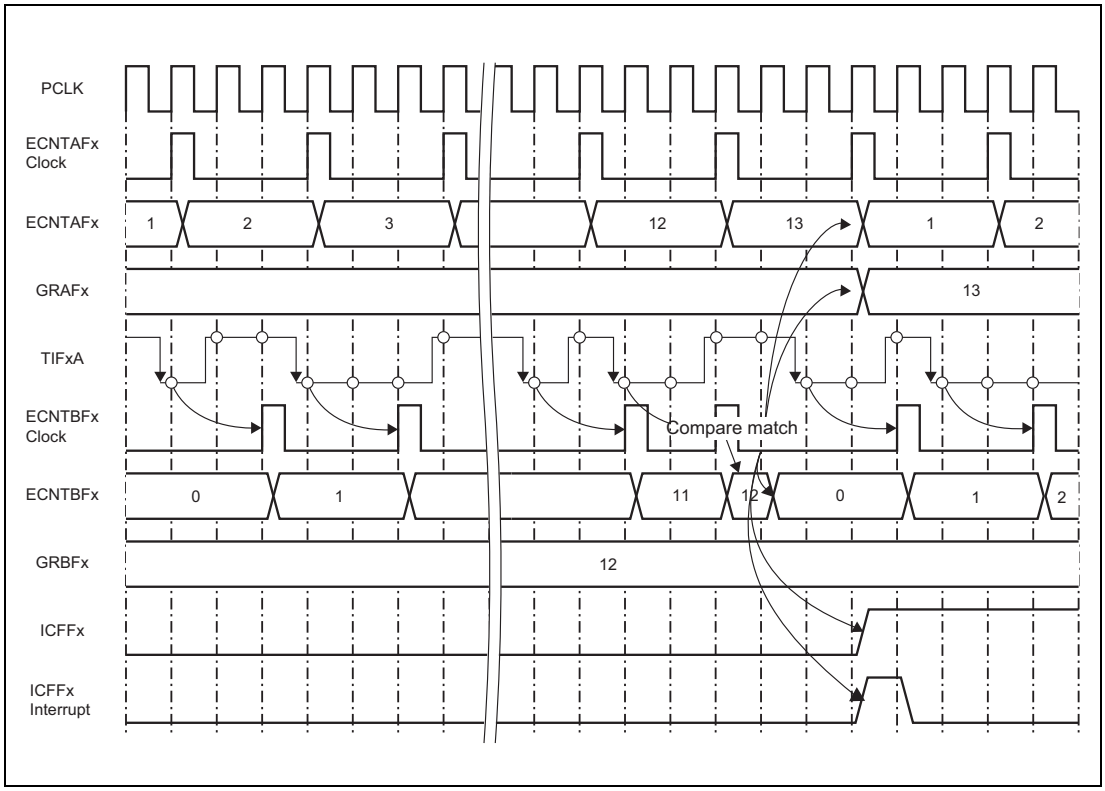


Figure 21.62 Operation Example of Effective Edge Interval Counting

21.9.3.3 Measurement of Time during High/Low Input Levels

Measures the time while TIFxA or TIFx is driven high or low. The time obtained is indicated using the ECNTAFx clock source as the standard. The width of the measurement time is specified to GRBFx in the form of the pulse number provided for TIFxA or TIFx (GRBFx value). Operation of the timer F is described below. **Figure 21.63** shows an operation example. This is the example in which the high level periods of the three pulses are measured as nine count source cycles. Timer counter ECNTAFx and event counter ECNTBFx are driven by the ECNTAFx and ECNTBFx clocks, respectively.

- ECNTAFx: Executes up-count using one of the clock buses 0 to 6 as a count source and TIFxA or TIFx level as enable. Therefore, the time period in which TIFxA or TIFx is in high level (EGSELFx = 2, falling edge selected), or TIFxA or TIFx is in low level (EGSELFx = 1, rising edge selected) is measured. After detecting a compare match in ECNTBFx, this register clears the count number synchronizing with the next count source clock. If TIFxA or TIFx is driven high (EGSELFx = 2) or low (EGSELFx = 1) at clearing count by the compare match, the count value becomes $0000\ 0001_H$. **Figure 21.64** is an example of this.
- ECNTBFx: Counts the falling edge of the signal from the TIFxA or TIFx input. Rising or falling edge is selectable to be counted. The operation example shows counting of the TIFxA or TIFx falling edge. A delay of two cycles occurs because of synchronization processing. When a compare match is detected, the count number is cleared synchronized with the next PCLK.
- GRAFx: Functions as the capture register for ECNTAFx. When a compare match in ECNTBFx is detected, this register captures the ECNTAFx count number synchronizing with the next ECNTAFx clock.
- GRBFx: Functions as the compare match register for ECNTBFx. A compare match is detected when the count number in ECNTBFx and GRBFx agree.
- Compare match interrupt request: After a compare match detection of ECNTBFx, a compare match interrupt request is issued to CPU at the next “ECNTAFx Clock” cycle.
- DMA transfer request output: After a compare match detection of ECNTBFx, a DMA transfer request is issued to DMAC at the next “ECNTAFx Clock” cycle.
- ICFFx: After detecting a compare match in ECNTBFx, sets the ICFFx flag synchronized with the next ECNTAFx clock.
- ECNTCFx, GRCFx, GRDFx: Do not function.

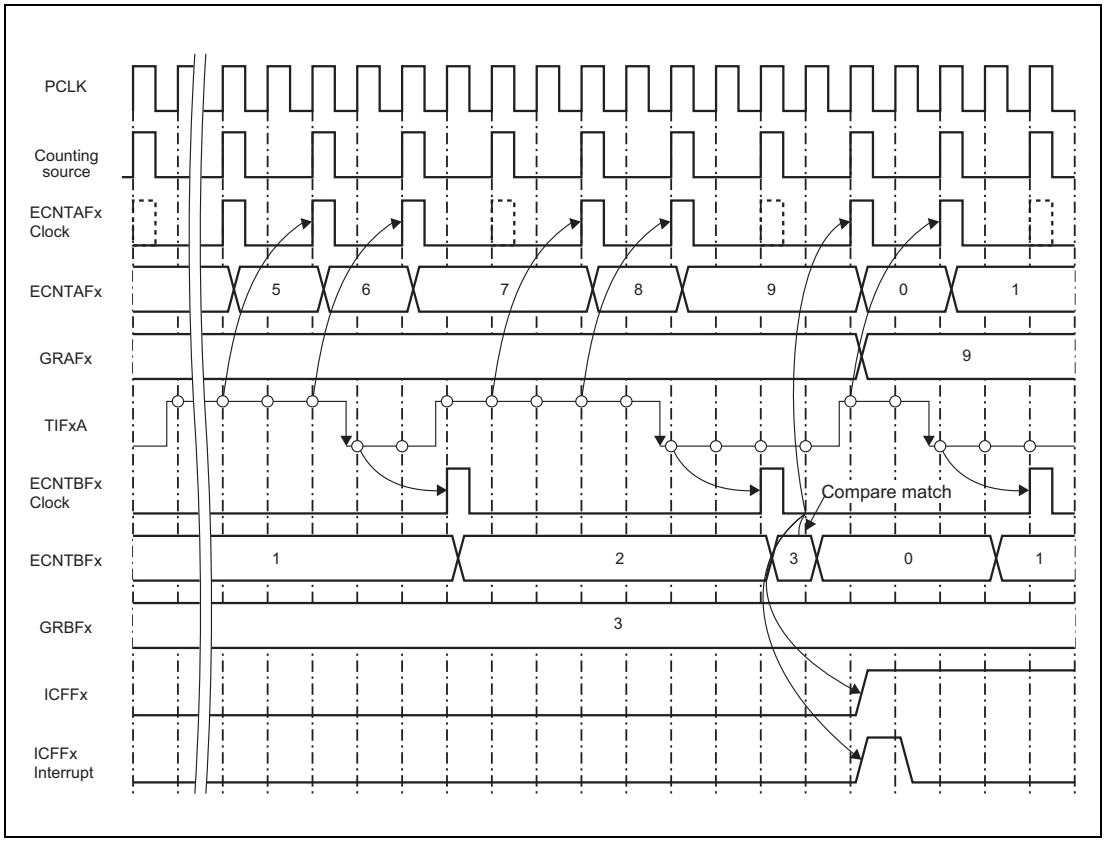


Figure 21.63 Operation Example of Measurement of Time during High Input Levels

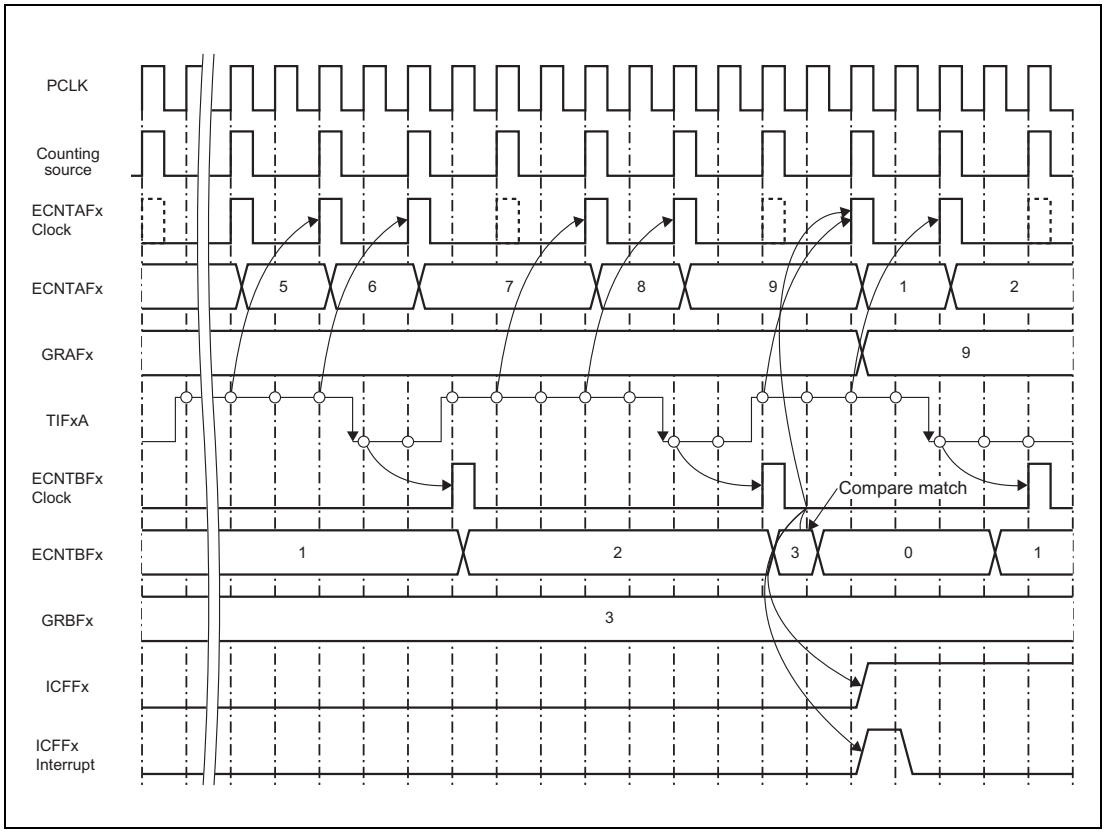


Figure 21.64 Operation Example of Measurement of Time during High Input Levels (TIFxA is in High Level When Capture is in Operation)

21.9.3.4 Measurement of PWM Input Waveform Timing

Measures the off-duty (non-active) period and cycle time of the PWM waveform input to TIFxA or TIFx. The off-duty period is measured as the period of either the high or low level input on TIFxA or TIFx, and the PWM cycle is measured as the interval between two rising or falling edges. Both are measured concurrently. The measured time is expressed in the number of cycles of the clock source for ECNTAFx. The duration of the measurement is set in GRBFx, which is specified as the number of PWM pulses input to TIFxA or TIFx.

Compare match is possible for the number of PWM cycles. A compare match can issue an interrupt request to CPU.

Operation of timer F is described below. **Figure 21.65** shows an operation example. This is the example in which two PWM cycles in PWM waveform are measured as six counter clock cycles and the off-duty period (low-level period) is measured as four counter clock cycles.

The clocks for ECNTAFx, ECNTBFx, and ECNTCFx in this example provide the timing of counting or clearing operation of the time counter ECNTAFx, event counter ECNTBFx, and ECNTCFx, respectively.

- ECNTAFx: Executes up-count using one of the clock buses 0 to 6 as a count source and TIFxA or TIFx input level as enable. Therefore, the time period in which TIFxA or TIFx is in high level (EGSELFx = 2, falling edge selected), or TIFxA or TIFx is in low level (EGSELFx = 1, rising edge selected) is measured. After detecting a compare match in ECNTBFx, this register clears the count number synchronizing with the next count source clock. If TIFxA or TIFx is driven high (EGSELFx = 2) or low (EGSELFx = 1) at clearing count by the compare match, the count value becomes $0000\ 0001_H$. The operation example shows measurement of low level.
- ECNTBFx: Counts the edge of the signal from the TIFxA or TIFx input. Rising or falling edge is selectable to be counted. The operation example shows counting of the rising edge. A delay of two cycles occurs because of synchronization processing. When a compare match is detected, the count number is cleared synchronized with the next PCLK.
- GRAFx: Functions as the capture register for ECNTAFx. When a compare match in ECNTBFx is detected, this register captures the ECNTAFx count number synchronizing with the next ECNTAFx clock.
- GRBFx: Functions as the compare match register for ECNTBFx. A compare match is detected when the count number in ECNTBFx and GRBFx agree.
- ECNTCFx: Measures time using the same count source as ECNTAFx. This register clears the count number synchronizing with the next ECNTAFx clock after detecting a compare match in ECNTBFx. Since ECNTCFx count clear occurs at the same time with countup, the cleared value is $0000\ 0001_H$. A compare match with GRDFx does not clear ECNTCFx, and the counting continues.
- GRCFx: Functions as the capture register for ECNTCFx. This register captures the ECNTCFx count number synchronizing with the next ECNTAFx clock after detecting a compare match in ECNTBFx.
- Input capture interrupt request: After a compare match detection of ECNTBFx, an input capture interrupt request is issued to CPU at the next “ECNTAFx Clock” cycle.
- DMA transfer request output: After a compare match detection of ECNTBFx, a DMA transfer request is issued to DMAC at the next “ECNTAFx Clock” cycle.
- ICFFx: After detecting a compare match in ECNTBFx, sets the ICFFx synchronized with the next ECNTAFx clock.

- GRDFx: This register works as a compare match register of ECNTCFx. A compare match occurs when the ECNTCFx count coincides with GRDFx. (Effective when the GRDFCMEN bit of the private function control register F (PVFCRF) is 1.)
- Overflow interrupt request output: After a compare match detection of ECNTCFx, an overflow interrupt request is issued to CPU at the next PCLK cycle. (Effective when the GRDFCMEN bit of the private function control register F (PVFCRF) is 1.)
- OVFCFx: After detection of a compare match of ECNTCFx, OVFCFx is set at the next PCLK cycle. (Effective when the GRDFCMEN bit of the private function control register F (PVFCRF) is 1.)

Therefore, ECNTBFx (GRBFx) and ECNTAFx (GRAFx) are operating in measurement of time during low input levels mode and ECNTBFx (GRBFx) and ECNTCFx (GRCFx) are operating in effective edge interval counting mode.

The measured values in PWM input waveform measurement mode are saved in two 32-bit registers: GRAFx and GRCFx. Be noted that, when reading these two registers, a capture might occur in the period between reading one register and reading the other register, resulting in reading incorrect measurements (for example, when an interrupt request was processed in between).

Reading values from backup registers would provide two consistent values. For details, see **Section 21.9.3.9, Simultaneous Access of Multiple Registers**.

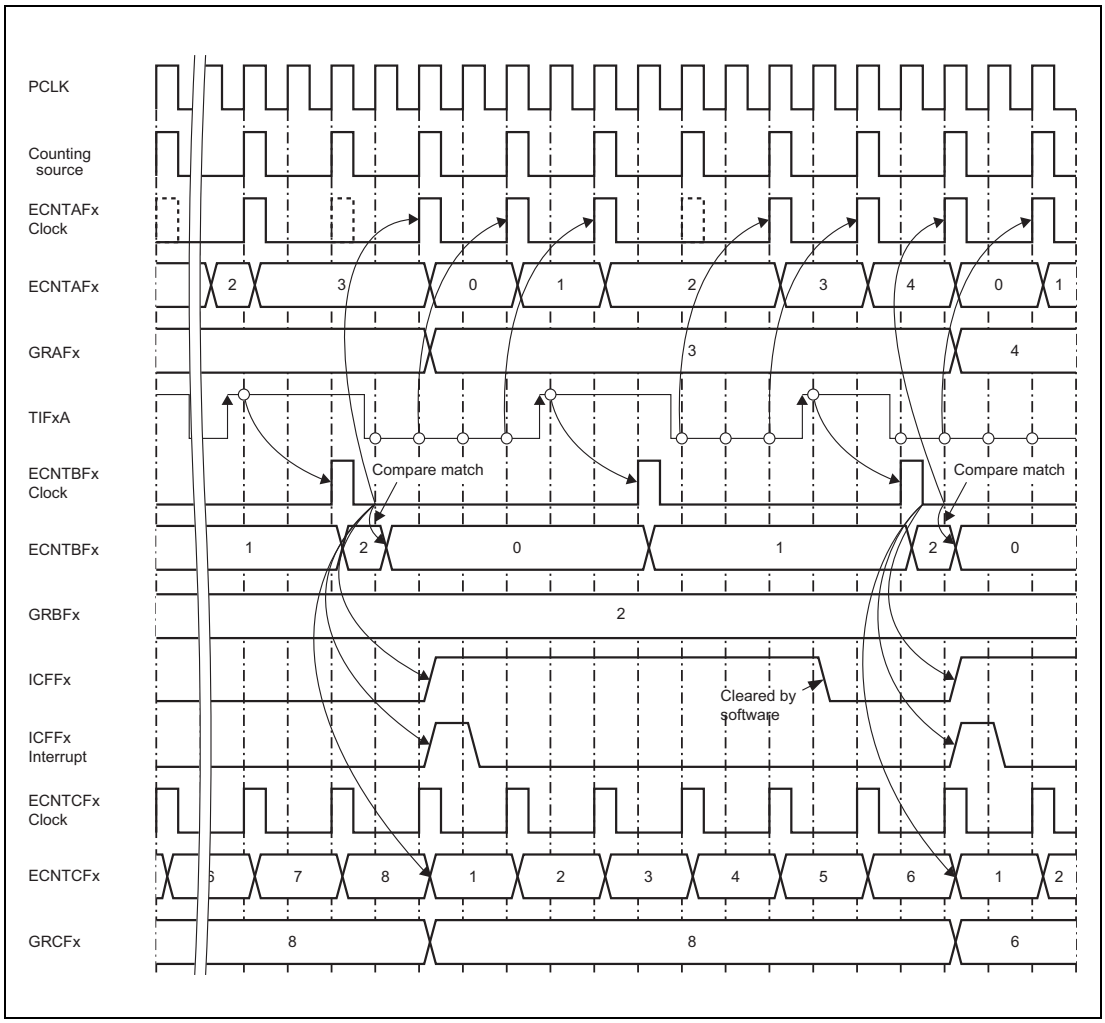


Figure 21.65 Operation Example of Measurement of PWM Input Waveform Timing

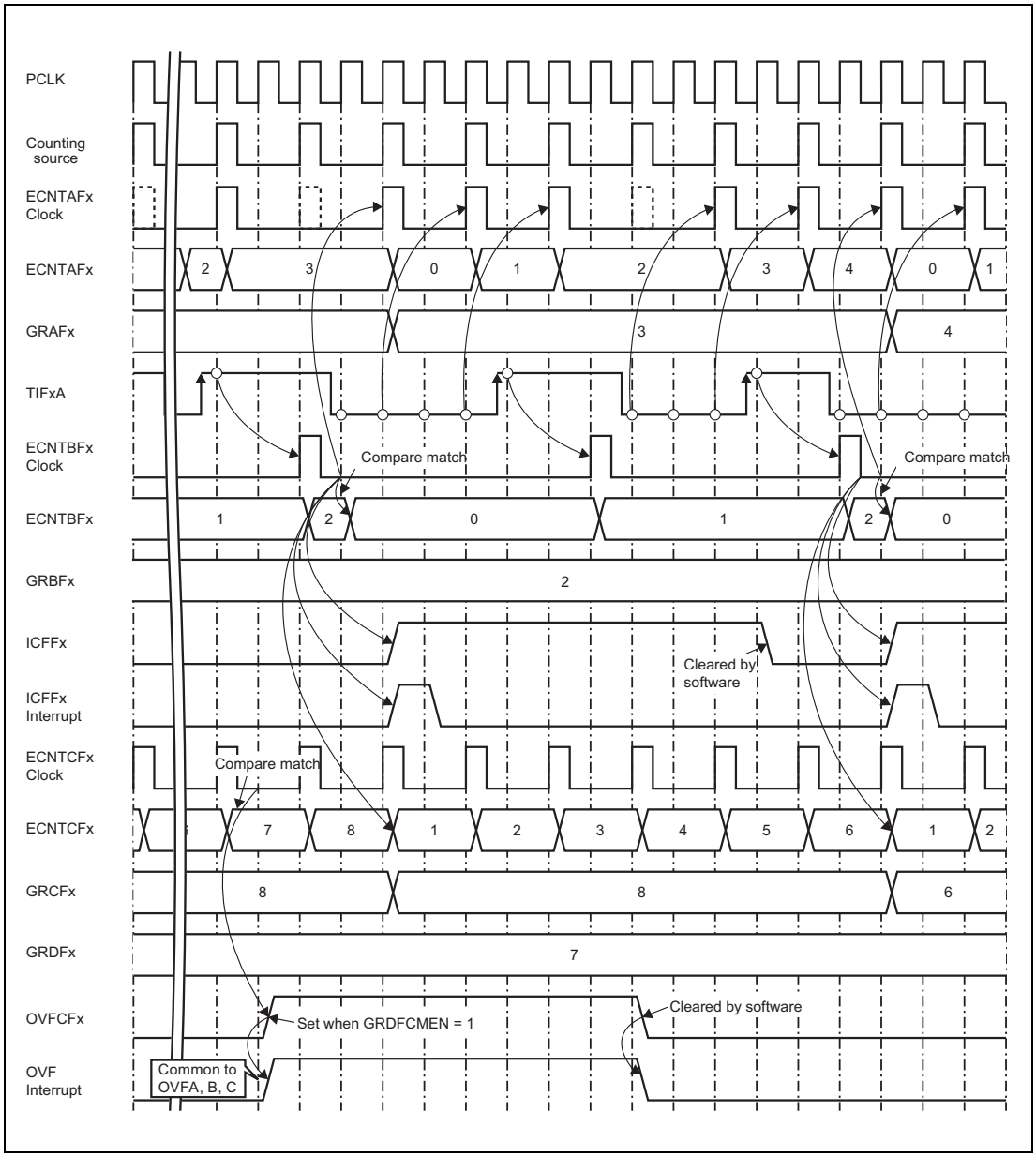


Figure 21.66 Operation Example of Measurement of PWM Input Waveform Timing

21.9.3.5 Rotation Speed/Pulse Measurement

Measures the number of edges input to TIFx, the edge input time (time stamp), the off-duty period in the PWM waveform that emerges between the last input edge and the edge this time, and PWM cycle time.

The time obtained is expressed using the ECNTAFx clock source as the standard. The maximum interval of edge input can be set to GRBFx, which enables to output an interrupt request if the edge input interval exceeds the maximum value.

At this moment, the timer F operates as shown below. **Figure 21.67** shows an example of operation. The ECNTAFx clock, ECNTBFx clock, and ECNTCFx clock show the timing of count operation or clearing for the time counter ECNTAFx, event counter ECNTBFx, and ECNTCFx, respectively.

- ECNTAFx: Executes up-count using one of the clock buses 0 to 6 as a count source and TIFx input level as enable. Therefore, the time period in which TIFx is in high level (EGSELFx = 2, falling edge selected), or TIFx is in low level (EGSELFx = 1, rising edge selected) is measured. After inputting the edge to TIFx, this register clears the count number synchronizing with the next count source clock. If TIFx is driven high (EGSELFx = 2) or low (EGSELFx = 1) at clearing count by the compare match, the count value becomes $0000\ 0001_H$.
- ECNTBFx: Counts the edge of the signal from the TIFx input. Rising or falling edge is selectable to be counted. The operation example shows counting of the falling edge. A delay of two cycles occurs because of synchronization processing.
- GRAFx: Functions as the capture register for ECNTAFx. This register captures the ECNTAFx count number synchronizing with the next ECNTAFx clock after inputting the edge to TIFx.
- GRBFx: Functions as the capture register for ECNTCFx. When the ECNTCFx count and the value in the 16 lower-order bits in GRBFx extended with 0 match, this register detects a compare match and set the OVFCFx to 1.
- ECNTCFx: Measures time using the same count source as ECNTAFx. This register clears the count number synchronizing with the next ECNTAFx clock after inputting the edge to TIFx. Since ECNTCFx count clear occurs in the same timing, the cleared value is $0000\ 0001_H$.
- GRCFx: Functions as the capture register for ECNTCFx. This register captures the ECNTCFx count number synchronizing with the next ECNTAFx clock after inputting the edge to TIFx.
- GRDFx: Functions as the capture register for ECNTCFx. This register captures the ECNTAFx, whose number being accumulated to GRDFx, synchronizing with the next ECNTAFx clock after inputting the edge to TIFx. The value to be added is the ECNTCFx value before clearing. (Capture value: GRDFx+ = ECNTCFx).
- Input capture interrupt request output: After an edge input to TIFx, an input capture interrupt request is issued to CPU at the next “ECNTAFx Clock” cycle.
- DMA transfer request output: After an edge input to TIFx, a DMA transfer request is issued to DMAC at the next “ECNTAFx Clock” cycle.
- ICFFx: Sets the ICFFx synchronizing with the next ECNTAFx clock after inputting the edge to TIFx.
- Overflow interrupt request output: An overflow interrupt request is issued to CPU at the next PCLK cycle during which values of ECNTCFx and GRBFx coincide (zero extension of 16 lower-order bits).
- OVFCFx flag: Sets the OVFCFx synchronizing with the next PCLK after the values in ECNTCFx and GRBFx (zero extension of 16 lower-order bits) match.

At the time of an input capture interrupt request, reading ECNTBFx, GRAFx, GRCFx, and GRDFx provides the number of edges, the off-state duty cycle, the PWM cycle, and the edge input time, respectively.

The capture timing of GRAFx, GRCFx, and GRDFx synchronizes with the count clock of ECNTAFx. Note that if the edge input cycle is shorter than the ECNTAFx count clock cycle, incorrect measurement may occur.

The measured values in “rotation speed/pulse measurement” mode are saved in three 32-bit registers GRAFx, GRCFx, and GRDFx. Be noted that, when reading these three registers, a capture might occur in the period between reading one register and reading another register, resulting in reading incorrect measurements (for example, when an interrupt request was processed in between).

Reading values from backup registers would provide three consistent values. For details, see **Section 21.9.3.9, Simultaneous Access of Multiple Registers**.

CAUTION

Do not set this mode for subblocks other than 03 to 15.

Figure 21.67 Operation Example of Rotation Speed/Pulse Measurement

21.9.3.6 Up/Down Event Count

This register uses the TIFxA pin, one of the two external input pins (TIFxA, TIFxB), as the count source, and TIFxB switches upcount to and from downcount. If a count period is designated to GRAFx the count number after designation can be obtained in GRBFx. The counting period is the period of ECNTAFx count source clock (GRAFx value).

At this moment, the timer F operates as shown below. **Figure 21.68** shows an example of operation. The ECNTAFx clock and ECNTBFx clock show the timing that time counter ECNTAFx and event counter ECNTBFx execute the count operation or clearing, respectively.

- ECNTAFx: Measures time using one of the clock buses 0 to 6. When a compare match is detected, the count value is cleared synchronized with the next PCLK.
- ECNTBFx: Upcount/downcount operation is performed at both rising and falling edges of TIFxA. Count direction is determined by the TIFxB input level. (See **Table 21.138**.) Because of synchronization processing, a delay of two cycles occurs in TIFxA and TIFxB.
- GRAFx: Functions as the compare match register for ECNTAFx. A compare match is detected when the count number in ECNTAFx and GRAFx agree.
- GRBFx: Functions as the capture register for ECNTBFx. When a compare match in ECNTAFx is detected, this register captures the ECNTBFx count number synchronizing with the next PCLK.
- Input capture interrupt request output: After compare match detection of ECNTAFx, an interrupt request is issued to CPU at the next PCLK cycle.
- DMA transfer request output: After compare match detection of ECNTAFx, a DMA transfer request is issued to DMAC at the next PCLK cycle.
- ICFFx: After detecting a compare match in ECNTAFx, sets the ICFFx flag synchronized with the next PCLK.
- ECNTCFx, GRCFx, GRDFx: Do not function

Table 21.138 Count Direction in Up/Down Event Count Mode

	Count Direction	
Input	**Upcount**	**Downcount**
TIFxA		
TIFxB	Low	High

CAUTION

Always set this mode only in subblock 0 to subblock 2.

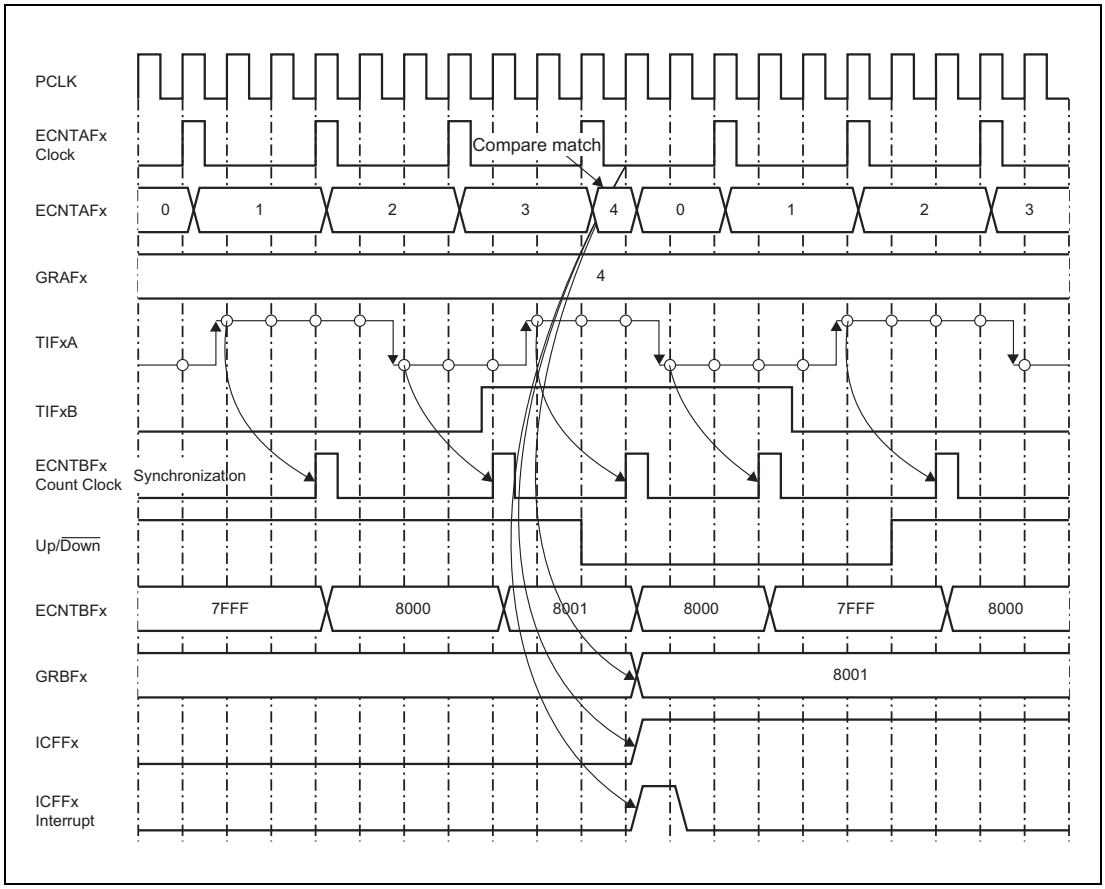


Figure 21.68 Operation Example of Up/Down Event Count

21.9.3.7 Four-time Multiplication Event Count

The count operation is executed using the external two input pins (TIFxA, TIFxB) as the count sources. Upcount or downcount is switched according to their input states. If a count period is designated to GRAFx, the count number after designation can be obtained in GRBFx. The counting period is the period of ECNTAFx count source clock (GRAFx value).

At this moment, the timer F operates as shown below. **Figure 21.69** shows an example of operation. The ECNTAFx clock and ECNTBFx clock show the timing that time counter ECNTAFx and event counter ECNTBFx execute the count operation or clearing, respectively.

- ECNTAFx: Measures time using one of the clock buses 0 to 6. When a compare match is detected, the count value is cleared synchronized with the next PCLK.
- ECNTBFx: Upcount/downcount operation is performed at both rising and falling edges of TIFxA and TIFxB respectively. Count direction is determined by the other signal input level. (See **Table 21.139**). Because of synchronization processing, a delay of two cycles occurs in TIFxA and TIFxB.
- GRAFx: Functions as the compare match register for ECNTAFx. A compare match is detected when the count number in ECNTAFx and GRAFx agree.
- GRBFx: Functions as the capture register for ECNTBFx. When a compare match in ECNTAFx is detected, this register captures the ECNTBFx count number synchronizing with the next PCLK.
- Input capture interrupt request output: After a compare match detection of ECNTAFx, an input capture interrupt request is issued to CPU at the next PCLK cycle.
- DMA transfer request output: After compare match detection of ECNTAFx, a DMA transfer request is issued to DMAC at the next PCLK cycle.
- ICFFx: After detecting a compare match in ECNTAFx, sets the ICFFx synchronized with the next PCLK.
- ECNTCFx, GRCFx, GRDFx: Do not function

Table 21.139 Count Direction in Four-time Multiplication Event Count Mode

Input	Count Direction							
	Upcount				Downcount			
TIFxA	High		Low		High		Low	
TIFxB		High		Low		Low		High

CAUTION

Always set this mode only in subblock 0 to subblock 2.

Note: Operation when edge inputs in TIFxA and TIFxB are detected simultaneously is not guaranteed. The interval between edge inputs in TIFnA and TIFnB must be at least 1.5 cycles (PCLK).

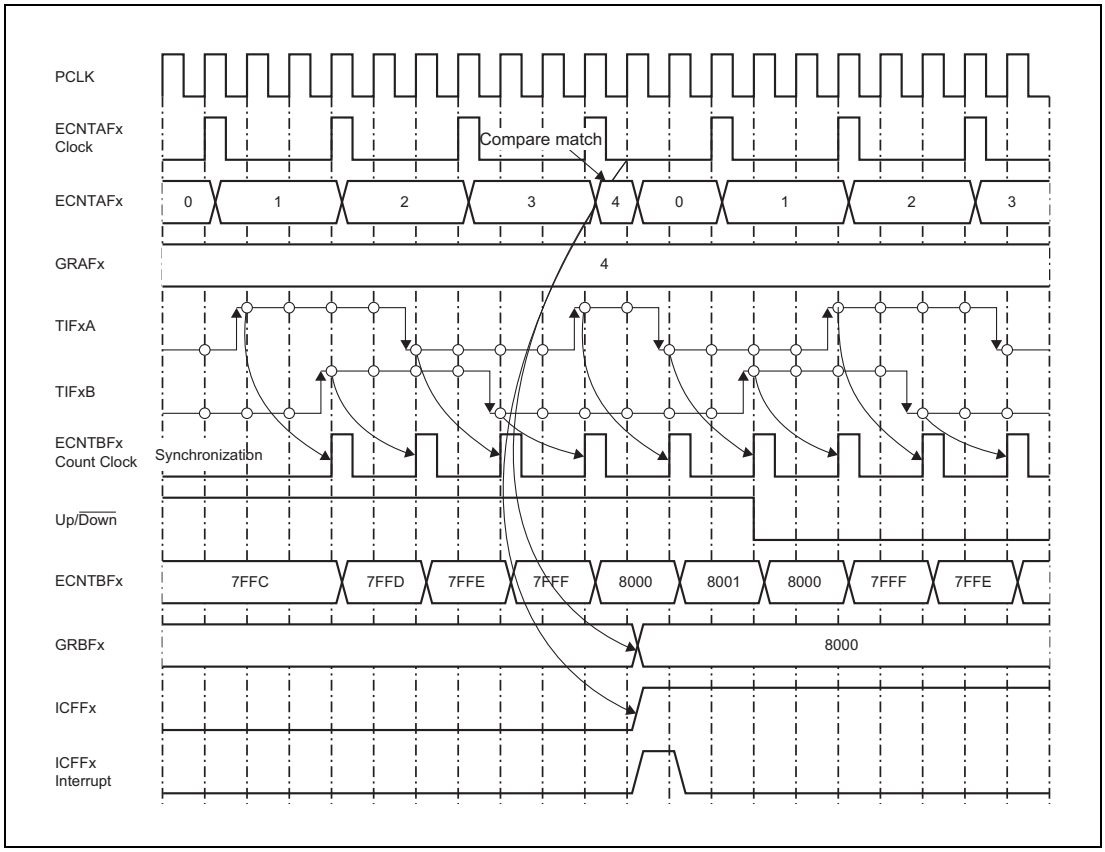


Figure 21.69 Operation Example of Four-time Multiplication Event Count Overflow/ Underflow

21.9.3.8 Overflow and Underflow

Counter clearing of FFFF $FFFF_H$ (ECNTAFx and ECNTCFx) to 0000 0000_H (ECNTAFx and ECNTCFx) or $FFFF_H$ (ECNTBFx) to 0000_H (ECNTBFx) causes overflow detection. In this case, the overflow flag is reset and an overflow interrupt request is issued as soon as the counter value becomes 0000 0000_H (or 0000_H). (A figure is omitted.) When an overflow of ECNTAFx is detected, OVFAFx is set and an overflow A interrupt request is issued. When an overflow of ECNTBFx is detected, OVFBFx is set and an overflow B interrupt request is issued. When an overflow of ECNTCFx is detected, OVFCFx is set and an overflow C interrupt request is issued.

An underflow is detected when the counter value changes from 0000_H (ECNTBFx) to $FFFF_H$ (ECNTBFx). In this case, OVFBFx is reset and an overflow interrupt request is issued as soon as the counter value becomes $FFFF_H$. (A figure is omitted.) OVFBFx is also set. An underflow occurs only with ECNTBFx, in which case an overflow B interrupt request is issued.

21.9.3.9 Simultaneous Access of Multiple Registers

In PWM input waveform measurement mode and rotation speed/pulse measurement mode, saving register values of measurement results to backup registers provides multiple register values of the same time point. When the CDRFx register is read, register values are saved from GRAFx to BGRAFx, from GRCFx to BGRCFx, and from GRDFx to BGRDFx.

For saving general register values to backup registers and for reading data from backup registers, the backup control register Fx (BKCRFx) must be set. For details, see the section above explaining backup control registers.

With the above method in PWM input waveform measurement mode, reading CDRFx and then BGRCFx (in the order of DMAC transfer access) provides the PWM cycle and the duty time of the same time point. In “rotation speed/pulse measurement” mode, reading CDRFx and then BGRAFx, BGRCFx, and BGRDFx provides data of the same time point.

21.10 Timer G

21.10.1 Operation Overview

The timer G block consists of 10 timer G subblocks.

A timer G subblock reads the input clock, and when a predetermined time period has passed, generates a pulse signal of one PCLK cycle period. The generated signals can be used as activation of the AD conversion device and interrupt request triggers. By a DMAC setting, the interrupt requests can be used as triggers of DMA transfer. An input clock can be selected for the counter from the seven clocks on the clock bus. The counter value of free-run counter TCNTG0 can be output as reference data for APA. For selection of output values, see **Section 22.5.8, APA Input Selector**.

Configuration

Timer G subblock includes one 32-bit counter (TCNTGx), one compare match register (OCRGx), and a controller.

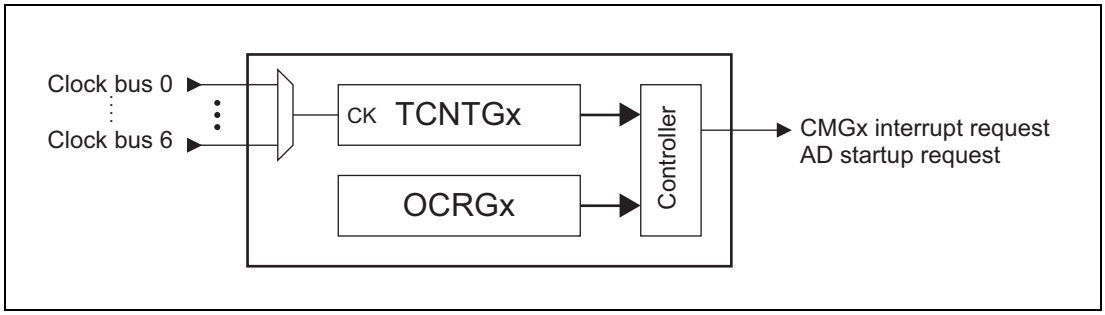


Figure 21.70 Timer G Configuration

Interrupt request

Timer G can issue 10 interrupt requests as described below.

- An interrupt request is issued upon compare match detection at timer subblock Gx. These requests are received by AD, DMAC, and the interrupt request controller, and they perform their processing according to individual settings.

21.10.2 Registers Related to Timer G

21.10.2.1 TSTRG — Timer Start Register G

Access: 8-bit /16-bit access is possible.

Address: FFE6 3900_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	STRG9	STRG8	STRG7	STRG6	STRG5	STRG4	STRG3	STRG2	STRG1	STRG0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.140 TSTRG Register Contents

Bit Position	Bit Name	Function
15 to 10	—	Not used. These bits are always read as 0. When writing, always write 0.
9 to 0	STRG9 to STRG0	Counter G Start 0: TCNTGx is disabled 1: TCNTGx is enabled

Note: x is an integer from 0 to 9.

The timer start register G (TSTRG) is a 16-bit register.

The 8 higher- and lower-order bits can be read and written as a TSTRGH register and a TSTRGL register respectively. This register specifies whether to start or stop each subblock (timer G0 to timer G9) included in timer G. The counting is not performed if the TGE bit of the ATU-IV control register (ATUENR) is not enabled even if the start bit of timer G permits counting.

When TSTRG is reset, it is initialized to 0000_H.

(1) STRGx — Counter G Start

These bits enable and disable timer counter Gx (TCNTGx) (G0 to G9) in the subblocks.

When these bits are cleared to 0, TCNTGx is stopped. While TCNTGx is stopped, it retains the previous value. When these bits are set to 1, TCNTGx is resumed from the previous value. TCNTGx runs when these bits and the TGE bit in ATUENR are both set to 1.

CAUTION

The prescalers run regardless of the counter G start bit and are not synchronized with the timing at which TCNTG is started. Therefore, the time from when the counter G start bit is set to when TCNTG is incremented for the first time is less than the cycle of the clock of TCNTG.

21.10.2.2 TCRGx — Timer Control Register Gx

(x = 0, 1, 2, 3, 4, 5, 6, 7, 8, 9: Corresponding to subblocks G0 to G9)

Access: 8-bit access is possible.

Address: FFE6 3910_H + (10_H * x)

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	CKSELGx[2:0]			—	—	—	—
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R	R	R	R

Table 21.141 TCRGx Register Contents

Bit Position	Bit Name	Function
7	—	Not used. This bit is always read as 0. When writing, always write 0.
6 to 4	CKSELGx[2:0]	Clock Select Gn These bits select the clock source of timer counter Gx (TCNTGn) of subblock. 000: Clock-bus line 0 001: Clock-bus line 1 010: Clock-bus line 2 011: Clock-bus line 3 100: Clock-bus line 4 101: Clock-bus line 5 110: Clock-bus line 6 111: Reserved
3 to 0	—	Not used. These bits are always read as 0. When writing, always write 0.

Timer control registers Gx (TCRG0 to TCRG9) are 8-bit readable/writable registers that set the operating mode of each subblock of timer G (G0 to G9).

When TCRG0 to TCRG9 are reset, they are initialized to 00_H.

(1) CKSELGx — Clock Select Gx

These bits select the clock source of timer counter Gx (TCNTGx) of subblocks (G0 to G9). Setting these bits to a value from 000 to 110 selects a clock source from clock bus 0 to clock bus 6.

However, do not set to 111. If set, operation cannot guaranteed.

21.10.2.3 TSRGx — Timer Status Registers Gx

(x = 0, 1, 2, 3, 4, 5, 6, 7, 8, 9: Corresponding to subblocks G0 to G9)

Access: 8-bit access is possible.

Address: FFE6 3912_H + (10_H * x)

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	OVFGx	CMFGx
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 21.142 TSRGx Register Contents

Bit Position	Bit Name	Function
7 to 2	—	Not used. These bits are always read as 0. When writing, always write 0.
1	OVFGx	Overflow Flag Gx 1: TCNTGx has overflowed 0: TCNTGx has not overflowed
0	CMFGx	Compare Match Flag Gx 1: Compare match has occurred in subblocks Gx 0: Compare match has not occurred in subblocks Gx

Timer status registers Gx (TSRG0 to TSRG9) are 8-bit readable/writable registers that measure the time and indicate occurrence of event counter overflow and compare match.

The overflow flag is the status flag for indicating the occurrence of overflow. It does not generate interrupts. The compare match flag is a status flag that indicates occurrence of an interrupt request. Setting a bit of the timer status clear register Gx (TSCRGx) clears the corresponding flag. If an interrupt source takes effect when this flag is set, an interrupt request is made again. An interrupt request is issued even in cases of contention between clearing by the corresponding timer status clear register and setting by an interrupt source.

When TSRG0 to TSRG9 are reset, they are initialized to 00_H.

(1) OVFGx — Overflow Flag Gx

Indicates whether or not timer counter Gx (TCNTGx) has overflowed. This bit cannot be set to 1 or 0 by software.

- Setting condition
  When TCNTGx overflowed (from FFFF $FFFF_H$ to 0000 0000_H)
- Clearing condition
  Writing 1 to OVFCGx in timer status clear register (Gx (TSCRGX))

(2) CMFGx — Compare Match Flag Gx

Indicates whether or not a compare match has occurred in subblocks Gx (G0 to G9). This flag cannot be set to 1 or 0 by software.

- Setting condition
  When compare match occurred in subblocks Gx
- Clearing conditions
  When 1 is written to CMFCGx of the timer status clear register Gx (TSCRGx).

21.10.2.4 TSCRGx — Timer Status Clear Register Gx

(x = 0, 1, 2, 3, 4, 5, 6, 7, 8, 9: Corresponding to subblocks G0 to G9)

Access: 8-bit access is possible.

Address: FFE6 3913_H + (10_H * x)

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	OVFCGx	CMFCGx
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Note 1. Value 0 cannot be written. The value is not retained when 1 is written. These bits are always read as 0.

Table 21.143 TSCRGx Register Contents

Bit Position	Bit Name	Function
7 to 2	—	Not used. These bits are always read as 0. When writing, always write 0.
1	OVFCGx	Overflow Flag Clear Gx Enable 0: Disabled (Initial value) 1: Writes 0 to OVFGx
0	CMFCGx	Compare Match Flag Clear Gx Enable 0: Disabled (Initial value) 1: Writes 0 to CMFGx

TSCRG is an 8-bit readable/writable register, which clears flags when compare match occur.

TSCRG is readable/writable only in 8-bit units. This register is always read as 0.

When TSCRG is reset, it is initialized to 00_H.

(1) OVFCGx — Overflow Flag Clear Gx Enable

When overflow flag Gx (OVFGx) of timer status register Gx (TSRGx) is set to 1, writing 1 to this register clears OVFGx to 0. This register is always read as 0.

(2) CMFCGx — Compare Match Flag Clear Gx Enable

When compare match flag Gx (CMFGx) of timer status register Gx (TSRGx) is set to 1, writing 1 to this register clears CMFGx to 0. This register is always read as 0.

21.10.2.5 TCNTGx — Timer Counters Gx

(x = 0, 1, 2, 3, 4, 5, 6, 7, 8, 9: Corresponding to subblocks G0 to G9)

Access: 32-bit access is possible. 8-bit access and 16-bit access are prohibited.

Address: FFE6 3914_H + (10_H * x)

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TCNTGx[31:16]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TCNTGx[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.144 TCNTGx Register Contents

Bit Position	Bit Name	Function
31 to 0	TCNTGx[31:0]	Timer Counter Gx These bits store the up-counter value.

Timer counters Gx (TCNTGx) are 32-bit readable/writable registers. These registers are provided one for each subblock and are incremented by the clock selected in the corresponding control register. Lines 0 to 6 of the clock bus can be selected as the input clock.

These counter values are compared with the value in compare match register Gx (OCRGx). When they match, Compare match flag (CMFGx bit in TSRGx register) is set and the counter value is cleared to 0000 0000_H in the next PCLK cycle. If counter clearing by compare match and incrementation occur simultaneously, TCNTGx is initialized to 0000 0001_H. This occurs when TCNTGx is driven by the clock whose frequency is equal to the PCLK. An interrupt request can be issued to CPU by detection of a compare match.

When TCNTGx is reset, it is initialized to 0000 0000_H.

21.10.2.6 OCRGx — Compare Match Registers Gx

(x = 0, 1, 2, 3, 4, 5, 6, 7, 8, 9: Corresponding to subblocks G0 to G9)

Access: 32-bit access is possible. 8-bit access and 16-bit access are prohibited.

Address: FFE6 3918$_H$ + (10$_H$ * x)

Value after reset: FFFF FFFF$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	OCRGx[31:0]															
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OCRGx[15:0]															
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.145 OCRGx Register Contents

Bit Position	Bit Name	Function
31 to 0	OCRGx[31:0]	Compare Match Gx These bits set the compare match value.

Compare match registers Gx (OCRGx) are 32-bit readable/writable registers. These registers are provided one for each subblock and function as the output compare register for timer counter Gx. A compare match can trigger an interrupt request to CPU or a DMA transfer request.

Do not set OCRGx to 0000 0000$_H$. If 0000 0000$_H$ is set, compare matches occur at unwanted cycles.

When OCRGx is reset, it is initialized to FFFF FFFF$_H$.

21.10.3 Detailed Operation Description

Specifying a time period in OCRGx generates a positive logical pulse of one PCLK cycle when the specified period has passed. The initial value of the output signal is 0. The counting period is specified based on the clock source of TCNTGx. The generated signals can be output to the AD conversion device for AD activation and interrupt request triggers.

The compare match flag Gx (CMFGx) of the timer status register Gx (TSRGx) is set when a compare match occurs. Compare match can trigger a DMA activation request to DMAC and an interrupt request to CPU.

Figure 21.71 illustrates the operation example. “TCNTGx Clock” here is an imaginary signal indicating the timing of counting and clearing of counter TCNTGx.

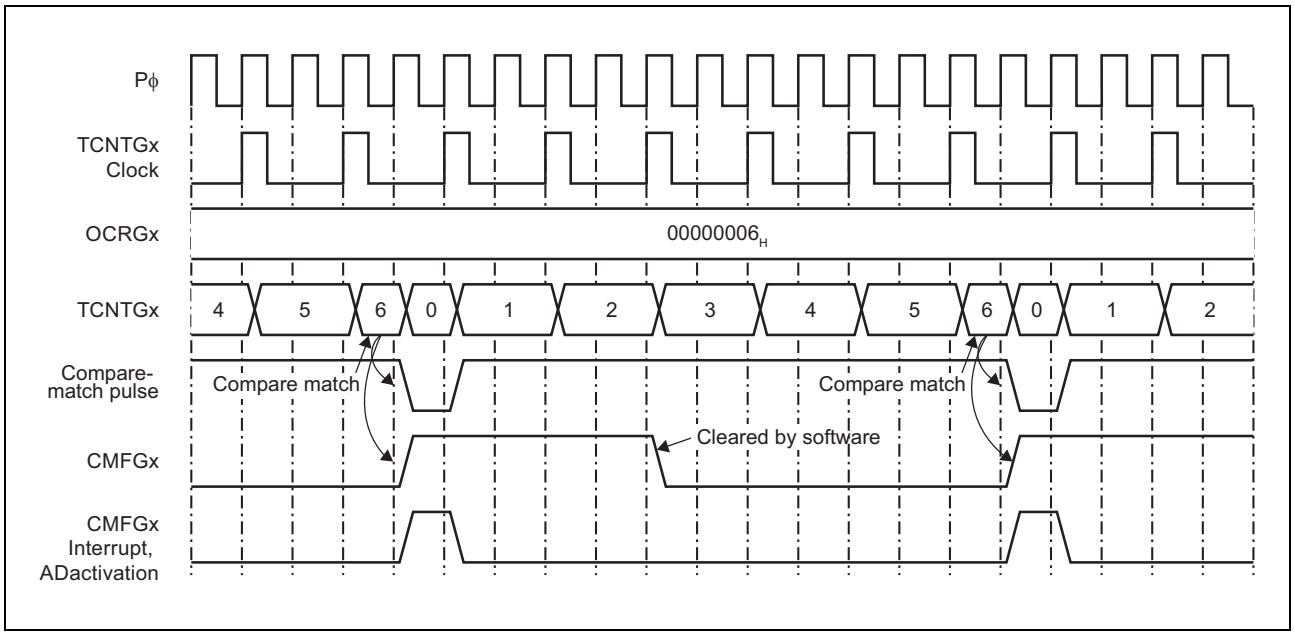


Figure 21.71 Operation Example of Counter and Compare Match

21.11 Timer H

21.11.1 Operation Overview

Timer H is a counter that measures a given time repeatedly.

Timer counter 1H (TCNT1H) is a 16-bit up-counter driven by the clock selected from seven lines of the clock bus. When the value in TCNT1H reaches the value in compare register 1H (OCR1H), it is cleared to 0000_H and is incremented again.

Timer counter 2H (TCNT2H) is a 32-bit counter that is incremented on compare match between TCNT1H and OCR1H, meaning that the counter counts the compare match.

Interrupt requests can be issued when TCNT2H is incremented.

Configuration

Timer H consists of one 16-bit timer counter 1H (TCNT1H), one compare match register 1_H (OCR1H), one 32-bit timer counter 2H (TCNT2H), and controller.

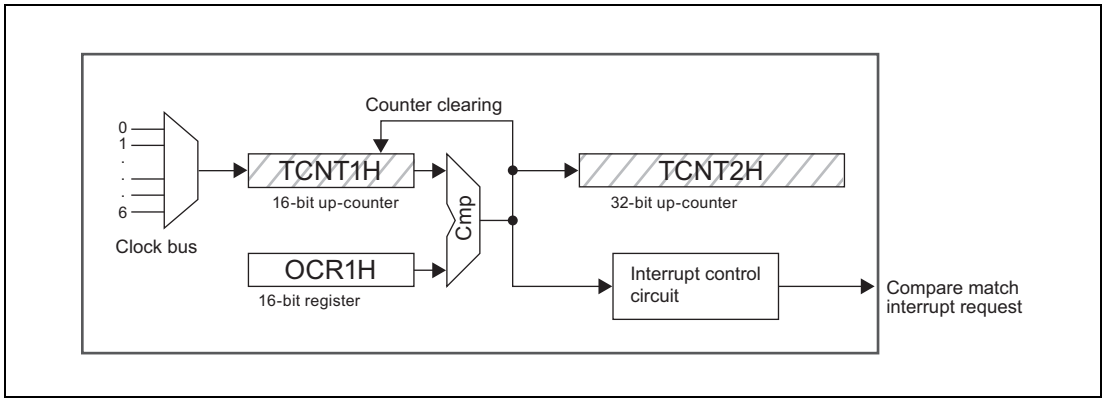


Figure 21.72 Block Diagram of Timer H Configuration

Interrupt request

Timer H can issue 1 interrupt requests as described below.

- Compare match interrupt request (1): Interrupt requests are issued on compare match between TCNT1H and OCR1H.

21.11.2 Registers Related to Timer H

21.11.2.1 TCRH — Timer Control Register H

Access: 8-bit access is possible.

Address: FFE6 0100_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	CKSELH[2:0]			—	—	—	—
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R	R	R	R

Table 21.146 TCRH Register Contents

Bit Position	Bit Name	Function
7, 3, 2	—	Not used. This bit is always read as 0. When writing, always write 0.
6 to 4	CKSELH[2:0]	These bits select the source of the clock of 16-bit timer counter 1H (TCNT1H). 000: Clock-bus line 0 001: Clock-bus line 1 010: Clock-bus line 2 011: Clock-bus line 3 100: Clock-bus line 4 101: Clock-bus line 5 110: Clock-bus line 6 111: Reserved
3 to 0	—	Not used. These bits are always read as 0. When writing, always write 0.

Timer control register H (TCRH) is an 8-bit readable/writable register. This register selects a count source for a 16-bit timer counter 1H (TCNT1H).

When TCRH is reset, it is initialized to 00_H.

(1) CKSELH — Clock Select H

These bits select the source of the clock of 16-bit timer counter 1H (TCNT1H).

Setting these bits to a value from 000 to 110 selects a clock source from clock bus 0 to clock bus 6. Do not set these bits to 111. If these values are set, the timer might not work properly.

21.11.2.2 TSRH — Timer Status Register H

Access: 8-bit access is possible.

Address: FFE6 0102_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	OVF2H	OVF1H	CMFH
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 21.147 TSRH Register Contents

Bit Position	Bit Name	Function
7 to 3	—	Not used. These bits are always read as 0. When writing, always write 0.
2	OVF2H	Overflow Flag 2H 1: TCNT2H has overflowed 0: TCNT2H has not overflowed
1	OVF1H	Overflow Flag 1H 1: TCNT1H has overflowed 0: TCNT1H has not overflowed
0	CMFH	Compare Match Flag H 1: Compare match between TCNT1H and OCR1H has occurred 0: Compare match between TCNT1H and OCR1H has not occurred

The timer status register H (TSRH) is an 8-bit read-only register.

This register indicates compare match of a 16-bit timer counter 1H (TCNT1H) and a compare match register (OCR1H), overflow of TCNT1H, and overflow of a 32-bit timer counter (TCNT2H).

These flags are status flags indicating overflow and compare match, and can be cleared by setting the corresponding bits of the timer status clear register H (TSCRH). If an interrupt source takes effect when this flag is set, an interrupt request is made again. An interrupt request is issued even in cases of contention between clearing by the corresponding timer status clear register and setting by an interrupt source.

When TSRH is reset, it is initialized to 00_H.

(1) OVF2H — Overflow Flag 2H

Indicates whether or not timer counter 2H (TCNT2H) has overflowed. This flag cannot be set to 1 or 0 by software.

OVF2H	Function
0	[Clearing condition] When 1 is written to OVFC2H of timer status clear register H (TSCRH) (Initial value)
1	[Setting condition] TCNT2H has overflowed (from FFFF FFFF$_H$ to 0000 0000$_H$)

Writing 0000 0000$_H$ to TCNT2H or starting TCNT2H up from the initial value (0000 0000$_H$) has no effect on this bit. When writing a value to the counter and incrementation occur simultaneously while the counter is FFFF FFFF$_H$, the overflow flag is set to 1 but the counter value is changed to the written value instead of 0000 0000$_H$.

(2) OVF1H — Overflow Flag 1H

Indicates whether or not timer counter 1H (TCNT1H) has overflowed. This flag cannot be set to 1 or 0 by software.

OVF1H	Function
0	[Clearing condition] When 1 is written to OVFC1H of timer status clear register H (TSCRH) (Initial value)
1	[Setting condition] TCNT1H has overflowed (from FFFF$_H$ to 0000$_H$)

Writing 0000$_H$ to TCNT1H or starting TCNT1H up from the initial value (0000$_H$) has no effect on this bit When writing a value to the counter and incrementation occur simultaneously while the counter is FFFF, the overflow flag is set to 1 but the counter value is changed to the written value instead of 0000 0000$_H$.

(3) CMFH — Compare Match Flag H

Indicates whether or not compare match between TCNT1H and OCR1H has occurred. This flag cannot be set to 1 or 0 by software.

CMFH	Function
0	[Clearing condition] When 1 is written to CMFCH of timer status clear register H (TSCRH) (Initial value)
1	[Setting condition] When the values in TCNT1H and OCR1H match

While CMFH is set to 1, meaning that the flag has not been cleared, the next compare match can be detected. In this case, 1 is rewritten to this bit.

21.11.2.3 TSCRH — Timer Status Clear Register H

Access: 8-bit access is possible.
Value 0 cannot be written. The value is not retained when 1 is written. These bits are always read as 0.

Address: FFE6 0103_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	OVFC2H	OVFC1H	CMFCH
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W

Table 21.148 TSCRH Register Contents

Bit Position	Bit Name	Function
7 to 3	—	Not used. Fix these bits to 0.
2	OVFC2H	Overflow Flag Clear 2h Enable 0: Disabled (Initial value) 1: Writes 0 to OVF2H.
1	OVFC1H	Overflow Flag Clear 1h Enable 0: Disabled (Initial value) 1: Writes 0 to OVF1H.
0	CMFCH	Compare Match Flag Clear H Enable 0: Disabled (Initial value) 1: Writes 0 to CMFH.

TSCRH is an 8-bit readable/writable register, which clears flags when compare matches occur.

TSCRH is readable/writable only in 8-bit units. This register is always read as 0.

When TSCRH is reset, it is initialized to 00_H.

(1) OVFC2H — Overflow Flag Clear 2H Enable

When overflow flag 2H (OVF2H) of timer status register H (TSRH) is set to 1, writing 1 to this register clears OVF2H to 0. This register is always read as 0.

(2) OVFC1H — Overflow Flag Clear 1H Enable

When overflow flag 1H (OVF1H) of timer status register H (TSRH) is set to 1, writing 1 to this register clears OVF1H to 0. This register is always read as 0.

(3) CMFCH — Compare Match Clear H Enable

When compare match flag H (CMFH) of timer status register H (TSRH) is set to 1, writing 1 to this register clears CMFH to 0. This register is always read as 0.

21.11.2.4 TCNT1H — Timer Counter 1H

Access: 16-bit access is possible. 8-bit access is prohibited.

Address: FFE6 0104_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TCNT1H[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.149 TCNT1H Register Contents

Bit Position	Bit Name	Function
15 to 0	TCNT1H[15:0]	Timer Counter 1H 16-bit up-counter

TCNT1H is a 16-bit readable/writable register. 8-bit access is prohibited.

TCNT1H run when the THE bit in the ATU-IV master enable register (ATUENR) and is incremented by the clock selected in timer control register H (TCRH). Overflow flag 1H (OVF1H) in timer status register H (TSRH) is set to 1 when TCNT1H overflows (from $FFFF_H$ to 0000_H)

The value in TCNT1H is constantly compared with the value in compare match register 1H (OCR1H). When they match, the CMFH bit in TSRH is set to 1 and TCNT1H is cleared to 0000_H in the next PCLK cycle.

However, if counter clearing by compare match and incrementation occur simultaneously, TCNT1H is cleared to 0001_H. This occurs when TCNT1H is driven by the clock whose frequency is equal to the PCLK. An interrupt request can be issued to CPU by detection of a compare match.

At the same time as compare match, timer counter 2H (TCNT2H) is incremented.

When TCNT1H is reset, it is initialized to 0000_H.

21.11.2.5 OCR1H — Compare Match Register 1H

Access: 16-bit access is possible. 8-bit access is prohibited.

Address: FFE6 0106$_H$

Value after reset: FFFF$_H$

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OCR1H[15:0]															
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.150 OCR1H Register Contents

Bit Position	Bit Name	Function
15 to 0	OCR1H[15:0]	Timer Counter 1H These bits specify the compare match value.

OCR1H is a 16-bit readable/writable register that function as the output compare register for timer counter 1H (TCNT1H). Compare match occurs at the following cycles.

Cycle of compare match = cycle of the TCNT1H counter clock (selected in the CKSELH bit in TCRH) × value in OCR1H

TCNT2H is incremented at this cycle. An interrupt request is issued to CPU.

Do not set OCR1H to 0000$_H$. If set, compare match occurs at unwanted cycles.

When this register is reset, it is initialized to FFFF$_H$.

21.11.2.6 TCNT2H — Timer Counter 2H

Access: 32-bit access is supported. 8-bit access and 16-bit access are prohibited.

Address: FFE6 0108$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TCNT2H															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TCNT2H															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.151 TCNT2H Register Contents

Bit Position	Bit Name	Function
31 to 0	TCNT2H	Timer Counter 2H These bits store the 32-bit counter value.

Timer counter 2H (TCNT2H) is a 32-bit readable/writable register. Do not write or read 16-bit or 8-bit data for this register.

This register is incremented by compare match between a 16-bit timer counter 1H (TCNT1H) and a compare match register 1H (OCR1H).

TCNT2H runs when the THE bit in the ATU-IV master enable register (ATUENR).

When TCNT2H overflowed (FFFF FFFF$_H$ → 0000 0000$_H$), the overflow flag 2H (OVF2H) in the timer status register H (TSRH) is set to 1.

When TCNT2H is reset, it is initialized to 0000 0000$_H$.

21.11.3 Operation Description

Compare match occurs and the compare match flag (CMFH) is set to 1 when a time set in the compare match register 1H (OCR1H) has elapsed. At this time, the timer counter 2H (TCNT2H) is incremented and the timer counter 1H (TCNT1H) is cleared to 0000_H.

The TCNT1H counter clock is selected by TCRH. TCNT1H and TCNT2H run when the THE bit in ATUENR is set to 1. If the THE bit is cleared to 0 while the counters are in operation, TCNT1H and TCNT2H are stopped and retain the counter value unchanged. When the THE bit is set to 1, the counters are resumed from the retained value.

A compare match of compare match register 1H (OCR1H) can issue an interrupt request to CPU.

Figure 21.73 shows an operation example of timer H. In this example, the TCNT1H counter clock is an ideal signal to show counter operation or clearing operation timing.

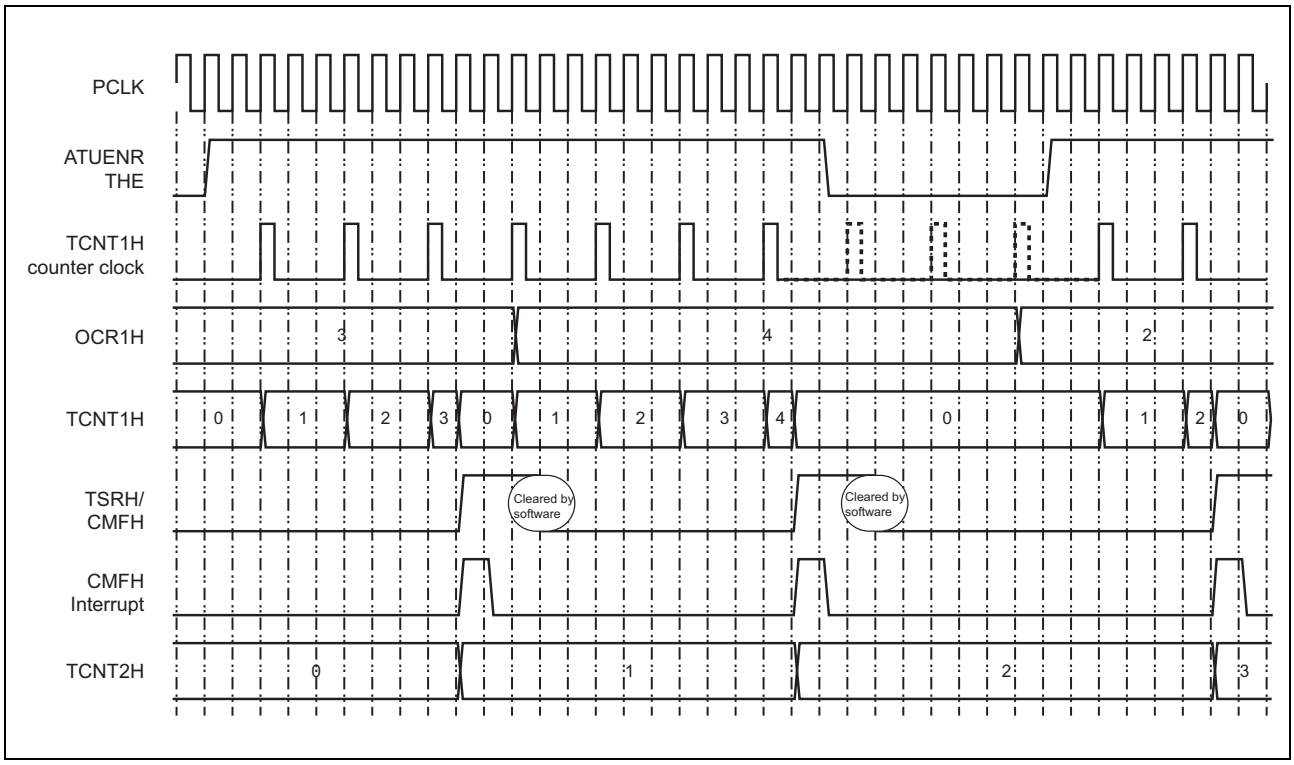


Figure 21.73 Operations of Timer H

21.12 Timer J

21.12.1 Operation Overview

The timer J block consists of 6 subblocks.

A timer J subblock counts the input clock to measure a predetermined time period repeatedly.

A 32-bit counter (TCNTJx) performs counting by a clock selected from seven clock buses. TCNTJx is cleared by the input edge of the TIJx pin. Timer J also has nine 32-bit FIFO registers which latch the value in TCNTJx every edge input. When the FIFO becomes full, DMAC activation or an interrupt request can be issued.

The validity period of the FIFO registers can be set to the period from a compare match of the compare match register (OCRJx) to the time when the FIFO registers become full.

Noise of the externally input signals (TIJx) can be removed by the input noise cancellation function.

Configuration

Timer J subblock includes one 32-bit counter (TCNTJx), one compare match register Jx (OCRJx), nine 32-bit FIFO registers, input signal controller (edge extractors and noise cancelers), and controller.

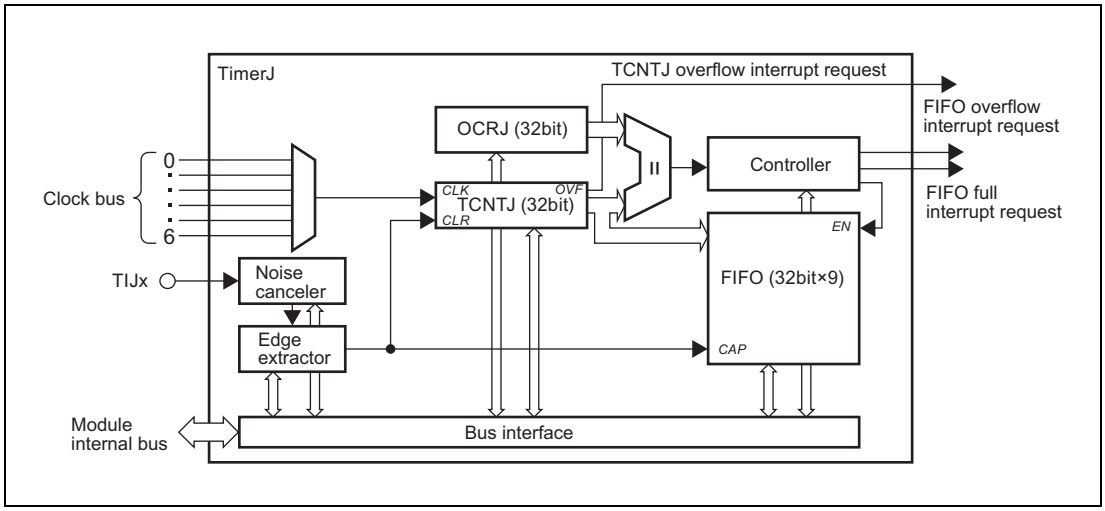


Figure 21.74 Block Diagram of Timer J

21.12.2 Registers Related to Timer J

21.12.2.1 TSTRJ — Timer Start Register J

Access: 8-bit access is possible.

Address: FFE6 3C00$_H$

Value after reset: 00$_H$

Bit	7	6	5	4	3	2	1	0
	—	—	STRJ5	STRJ4	STRJ3	STRJ2	STRJ1	STRJ0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.152 TSTRJ Register Contents

Bit Position	Bit Name	Function
7, 6	—	Not used. These bits are always read as 0. When writing, always write 0.
5 to 0	STRJ5 to STRJ0	Counter Jn Start 0: Counting of TCNTJn is stopped 1: Counting of TCNTJn is enabled

Note: x is an integer from 0 to 5.

Timer start register J (TSTRJ) is an 8-bit readable/writable register that enables and disables the subblocks (timer Jx) of timer J. Timer J counters run when the counter J start bit (STRJ) and timer J enable bit (TJE) in the ATU-IV master enable register (ATUENR) are both set to 1.

When TSTRJ is reset, it is initialized to 00$_H$.

(1) STRJx — Counter J Start

This bit specifies whether to enable or disable the timer counter Jx (TCNTJx) of timer subblock Jx.

When this bit is cleared to 0, TCNTJx stops operation. When stopped, the counter value is retained. Counting starts from the retained value when this bit is set to 1 again.

Even if the counter J start bit is set to 1, counting does not start unless the TJE bit of the ATU-IV enable register is set.

CAUTION

The prescalers run regardless of the counter J start bit and are not synchronized with the timing at which the TCNTJx is started. Therefore, the time from when the THE bit is set to when TCNTJx is incremented for the first time is less than the cycle of the clock of TCNTJx.

21.12.2.2 TCRJx — Timer Control Registers Jx

(x = 0, 1, 2, 3, 4, 5: Corresponding to subblocks J0 to J5)

Access: 8-bit access is possible.

Address: FFE6 3C20$_H$ + (20$_H$ * x)

Value after reset: 00$_H$

Bit	7	6	5	4	3	2	1	0
	—	CKSELJx[2:0]			—	NCEJx	IOJx	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R	R/W	R/W	R/W

Table 21.153 TCRJx Register Contents

Bit Position	Bit Name	Function
7, 3	—	Not used. These bits are always read as 0. When writing, always write 0.
6 to 4	CKSELJx	Clock Select Jx Specify the clock source of the timer counter Jx (TCNTJx). 000: Clock bus 0 001: Clock bus 1 010: Clock bus 2 011: Clock bus 3 100: Clock bus 4 101: Clock bus 5 110: Clock bus 6 111: Reserved
2	NCEJx	Noise Canceler Enable Jx 0: Disables the noise canceler of the TIJx input. 1: Enables the noise canceler of the TIJx input.
1, 0	IOJx	I/O Control Jx 00: Input capture of TIJx is disabled. 01: Input capture is made with TIJx rises. 10: Input capture is made with TIJx falls. 11: Input capture is made with TIJx rises and falls.

Timer control registers Jx (TCRJx) are 8-bit readable/writable registers that select operation modes of each subblock (timer Jx).

When TCRJ is reset, it is initialized to 00$_H$.

(1) CKSELJx — Clock Select Jx

These bits select the clock source of timer counter Jx (TCNTJx) of subblock. Setting these bits to a value from 000 to 110 selects a clock source from clock bus 0 to clock bus 6. However, do not set to 111. If set, operation cannot guaranteed.

(2) NCEJx — Noise Canceler Enable Jx

These bits enable and disable the noise cancelers for externally input signals (TIJx)

If a level change of external input TIJx is detected after this bit is set to 1, the setting of the noise cancel mode register (NCMR) of the common controller determines the operation mode to be started from the following modes: premature-transition cancellation mode, minimum time-at-level cancellation mode, and level accumulation cancellation mode.

- In premature-transition cancellation mode

When a level change of the externally input signal is detected, the change is output as the signal whose noise is removed and the corresponding noise canceler counter Jx (NCNTJx) is started for counting up. Subsequent level changes are masked until the value in the counter reaches the value in the noise canceler register Jx (NCRJx). The level of the externally input signal is output on this compare match.

When these bits are cleared to 0 while the counter (NCNTJx) is being incremented, counting continues until the values in the counter and the noise canceler register match. The subsequent level changes are also masked over this period.

- In minimum time-at-level cancellation mode

When a level change of the externally input signal is detected, the corresponding noise canceler counter Jx (NCNTJx) is started for counting up. If subsequent level changes are not detected until the value in the counter reaches the value in the noise canceler register Jx (NCRJx), the previously accepted level change is output as the signal whose noise is removed on compare match of the counter and noise canceler register. When the subsequent level change is detected before the values in the counter and noise canceler register match, all the changes are treated as noise. Therefore the signal whose noise is removed is not changed.

When these bits are cleared to 0 while the counter Jx (NCNTJx) is being incremented, counting continues until the values in the counter and the noise canceler register match or a level change on the externally input signal is detected.

In level accumulation cancellation mode, the input signal level determines the up/down counting of the corresponding noise cancel counters Jx (NCNTJx). Up-counting is made with high-level input, down-counting is made with low-level input. Up-count continues until a compare match between the noise cancel counter and noise cancel register Jx (NCRJx), and down-count continues until the noise cancel counter matches $*0000_H$. If an up-count results in a compare match (NCRJx), the noise canceler output is updated to 1. If a down-count results in a compare match (0000_H), the noise canceler output is updated to 0.

In minimum time-at-level cancellation mode and premature-transition cancellation mode, level change detection is always made by P regardless of the selected noise cancel clock. In level accumulation cancellation mode, the sampling of the input level is made by the noise canceler clock.

For details on operations in each mode, see **Figure 21.1, Figure 21.2** and **Figure 21.3**.

(3) IOJx — I/O Control Jx

These bits select edges to be detected for input capture triggers. When these bits are set to 00, input capture is not made. When they are set to 01, 10, or 11, detection of the selected edge transfers the value of timer counter Jx (TCNTJx) to the corresponding FIFO register Jx (FIFOJx).

Edges are extracted from the signal after noise removal.

When the noise canceler is disabled, the selected edge is simply extracted from the external inputs (TIJx). When the noise canceler is enabled, the selected edge is extracted from the signals after noise removal.

Edge extraction is synchronized with the PCLK. Make sure that the frequency of the PCLK is at least twice the frequency of the external input signal. Otherwise, edge extraction will not be performed correctly.

21.12.2.3 FCRJx — FIFO Control Registers Jx

(x = 0, 1, 2, 3, 4, 5: Corresponding to subblocks J0 to J5)

Access: 8-bit access is possible.

Address: FFE6 3C21$_H$ + (20$_H$ * x)

Value after reset: 00$_H$

Bit	7	6	5	4	3	2	1	0
	FIFOENJx	—	FVCRENJx	FRSTJx	—	—	FDFTRGJx	
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R	R/W	R/W*1	R	R	R/W	R/W

Note 1. Value 0 cannot be written. The value is not retained when 1 is written. These bits are always read as 0.

Table 21.154 FCRJx Register Contents

Bit Position	Bit Name	Function
6, 3, 2	—	Not used. These bits are always read as 0. When writing, always write 0.
7	FIFOENJx	FIFO Register Enable Jn Enables and disables FIFO register Jx (FIFOJx) 0: FIFO Jx is disabled 1: FIFO Jx is enabled
5	FVCRENJx	FIFO Enable Control Enable Jx Specifies whether to control the enabled period of FIFO registers. 0: FIFO is not enabled on OCRJx compare match 1: FIFO is enabled until it becomes full after OCRJx compare match
4	FRSTJx	FIFO Data Register Reset Jx Invalidates the data in the FIFO register, and resets it to the idle state. 0: No operation 1: FIFO is reset
1, 0	FDFTRGJx	FIFO Data Full Trigger Jx Specifies the number of data items that determines the setting of FDFFJx of the timer J status register. 00: 9 01: 6 10: 4 11: 2

FIFO control registers Jx (FCRJx) are 8-bit readable/writable registers that control FIFO of each subblock (timer J0 and timer J1). When FCRJx is reset, it is initialized to 00$_H$.

(1) FIFOENJx — FIFO Register Enable Jx

Enables and disables FIFO register Jx (FIFOJx) in subblocks J0 and J1. When this bit is set to 1 and the FIFO is disabled, even if an edge is input on pin TIJx, the value in TCNTJx is not latched in FIFO. However when FVCRENJx is set to 1, even if this bit (FIFOENJx) is 0, the value in TCNTJx is latched in the FIFO every input edge until FIFO becomes full (data count that fills FIFO full by the FDFTRGJx bit can be set) after compare match on OCRJx. When FVCRENJx = 1 and FIFOENJx = 1, the setting of FIFOENJx takes precedence.

(2) FVCRENJx — FIFO Enable Control Enable Jx

When FIFOENJx = 1 and FIFOENJx = 0, the FIFO is temporarily enabled until it becomes full (the threshold level can be set by FDFTRGJx) after compare match between timer counter Jx (TCNTJx) and compare match register Jx (OCRJx). Note that the FIFO is always enabled regardless of this bit when bit FIFOENJx is set to 1.

(3) FRSTJx — FIFO Data Register Reset Jx

When FRSTJx is set to 1, data capture by the FIFO buffer is disabled, after which the FIFO buffer is emptied and FDNRJx is reset while the FIFO is empty. However, the FDFFJx and FDOVFJx flags which have been set to 1 are not cleared. And the TSRJx.FVLDFJx is not also cleared.

(4) FDFTRGJx — FIFO Data Full Trigger Jx

These bits set the number of data words which is the threshold for setting the FDFFJx flag in the timer J status register Jx (TSRJx). When the FIFO is enabled and the number of data words in it has reached the trigger count, the FDFFJx bit is set to 1. When the FIFO is enabled on compare match between TCNTJx and OCRJx while FIFOENJx = 0 and FVCRENJx = 1, and then the number of data words has reached the trigger count, the FIFO is disabled.

CAUTION

When setting FCRJx to 20_H, read 9 captured values from the FIFOJx register during FIFO full interrupt handling, then reset the FIFO register (by writing 1 to the FRSTJx bit of the register) and clear the FIFO full interrupt request flag to 0*.

*** The procedure for clearing the FIFO full interrupt request flag to 0 is described below.**

1. **Write 0 to the FIFO full interrupt request flag of the interrupt controller (EIRFx bit in the EICx register).**
2. **Dummy-read the above register.**
3. **Execute an SYNCP instruction.**

21.12.2.4 TSRJx — Timer Status Register Jx

(x = 0, 1, 2, 3, 4, 5: Corresponding to subblocks J0 to J5)

Access: 8-bit access is possible.

Address: FFE6 3C22$_H$ + (20$_H$ * x)

Value after reset: 00$_H$

Bit	7	6	5	4	3	2	1	0
	—	—	—	FVLDFJx	CMFJx	OVFJx	FDOVFJx	FDFFJx
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 21.155 TSRJx Register Contents

Bit Position	Bit Name	Function
7 to 5	—	Not used. Fix these bits to 0.
4	FVLDFJx	FIFO Status Flag Jx 0: FIFO is not enabled 1: FIFO is enabled
3	CMFJx	Compare Match Flag Jx 0: Compare match between OCRJx and TCNTJx has not occurred 1: Compare match between OCRJx and TCNTJx has occurred
2	OVFJx	Overflow Flag Jx 0: TCNTJn has not overflowed 1: TCNTJn has overflowed
1	FDOVFJx	FIFO Data Overflow Flag Jx 0: Data in the FIFO is normal 1: Another capture has occurred while nine words of data was in the FIFO
0	FDFFJx	FIFO Data Full Flag Jx 0: Number of data words in FIFO is less than the value in FDFTRGJn 1: Number of data words in FIFO is equal to or exceeds the value in FDFTRGJn

Timer status registers Jx (TSRJx) are 8-bit readable/writable registers that indicate occurrence of overflow on timer counter Jx (TCNTJx), and compare match on compare register Jn (OCRJx). In addition, there are flags that indicate that the FIFO has overflowed and the number of data in the FIFO exceeds the trigger count.

The FIFO status flag and the compare match flag are status flags that indicate the status of FIFO and occurrence of timer compare matches. It does not generate interrupts. The overflow flag, FIFO data overflow flag, and FIFO data full flag are status flags for occurrence of interrupt requests. Setting a bit of the timer status clear register Jx (TSCRJx) clears the corresponding flag. If an interrupt source takes effect when this flag is set, an interrupt request is made again. An interrupt request is issued even in cases of contention between clearing by the corresponding timer status clear register and setting by an interrupt source.

When TSRJ is reset, it is initialized to 00$_H$.

(1) FVLDFJx — FIFO Status Flag Jx

Indicates whether the FIFO is enabled or disabled. This bit cannot be set or cleared by software.

When this bit is 1, the corresponding FIFO register Jx (FIFOJx) is enabled. The value in TCNTJx is latched in the FIFO on the edge of the TIJx pin.

When FIFOENJx = 1, this bit is always set to 1. When FIFOENJx = 0 and FVCRENJx = 1, FVLDFJx becomes 1 on compare match between TCNTJx and OCRJx. When the FIFO becomes full (matches the number of data set by FDFTRGJx), FVLDFJx is cleared to 0. When FIFOENJx = 0 and FVCRENJx = 0, this bit is always 0.

- Setting conditions

1. When FIFOENJx is set to 1
2. When FIFOENJx is 0, FVCRENJx is 1, and OCRJx compare match occurred

- Clearing conditions

1. When FIFOENJx is cleared to 0
2. When FIFOENJx is 0, FVCRENJx is 1, and the number of data words in FIFO exceeds the value set in FDFTRGJx

(2) CMFJx — Compare Match Flag Jx

Indicates that compare match between TCNTJx and OCRJx has occurred. When this bit is 1, compare match on compare match register Jx (OCRJx) has occurred. This bit cannot be set to 1 or 0 by software.

- Setting condition
  When the values in TCNTJx and OCRJx match
- Clearing condition
  When 1 is written to CMFCJx of timer status clear register Jx (TSCRJx).

(3) OVFJx — Overflow Flag Jx

Indicates that timer counter Jx (TCNTJx) has overflowed. When this bit is 1, timer counter Jx (TCNTJx) has overflowed. This bit cannot be set to 1 or 0 by software.

- Setting condition
  When TCNTJx has overflowed (from FFFF $FFFF_H$ to 0000 0000_H)
- Clearing condition
  When 1 is written to OVFCJx of the timer status clear register Jx (TSCRJx).

(4) FDOVFJx — FIFO Data Overflow Flag Jx

Indicates that another capture has occurred while nine words of data was in the FIFO. When this bit is 1, FIFOJx has overflowed and captured data is lost. This bit cannot be set to 1 or 0 by software.

- Setting condition
  When another capture has occurred while nine words of data was in the FIFO
- Clearing condition
  When 1 is written to FDOVFCJx of the timer status clear register Jx (TSCRJx).

(5) FDFFJx — FIFO Data Full Flag Jx

Indicates that the number of data words (TCNTJx counter values) captured in the FIFO is equal to or exceeds the value set in the FDFTRGJn bits in the FIFO control register Jx (FCRJx).

When this bit is 1, the number of data words captured in the FIFO is equal to or exceeds the set value and the captured data is ready to be read.

This bit cannot be set to 0 or 1 by software.

- Setting condition
  When the number of data words in FIFO has reached the value in FTRGJx
- Clearing condition
  When 1 is written to FDDFFCJx of the timer status clear register Jx (TSCRJx).
  However, the data in FIFOJn must be read to make the number of data words in the FIFO less than the value set in the FDFTRGJn bits.

21.12.2.5 TSCRJx — Timer Status Clear Register Jx

(x = 0, 1, 2, 3, 4, 5: Corresponding to subblocks J0 to J5)

Access: 8-bit access is possible.
Value 0 cannot be written. The value is not retained when 1 is written. These bits are always read as 0.

Address: FFE6 $3C23_H$ + (20_H * x)

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	CMFCJx	OVFCJx	FDOVFCJx	FDFFCJx
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 21.156 TSCRJx Register Contents

Bit Position	Bit Name	Function
7 to 4	—	Not used. Fix these bits to 0.
3	CMFCJx	Compare Match Flag Clear Jx Enable 0: Disabled (Initial value) 1: Writes 0 to CMFJx
2	OVFCJx	Overflow Flag Clear Jx Enable 0: Disabled (Initial value) 1: Writes 0 to OVFJx
1	FDOVFCJx	FIFO Overflow Flag Clear Jx Enable 0: Disabled (Initial value) 1: Writes 0 to FDOVFJx.
0	FDFFCJx	FIFO Table Full Flag Clear Jx Enable 0: Disabled (Initial value) 1: Writes 0 to FDFFJx.

Timer status clear register Jx (TSCRJx) is an 8-bit readable/writable register that specifies clearing of flags upon occurrence of compare match, overflow, overflow of FIFO registers, and data full.

TSCRJx is readable/writable only in 8-bit units. This register is always read as 0.

When TSCRJx is reset, it is initialized to 00_H.

(1) CMFCJx — Compare Match Flag Clear Jx Enable

When compare match flag Jx (CMFJx) of the timer status register Jx (TSRJx) is set to 1, writing 1 to this register clears CMFJx to 0. This register is always read as 0.

(2) OVFCJx — Overflow Flag Clear Jx Enable

When overflow flag Jx (OVFJx) of the timer status register Jx (TSRJx) is set to 1, writing 1 to this register clears OVFJx to 0. This register is always read as 0.

(3) FDOVFCJx — FIFO Overflow Flag Clear Jx Enable

When FIFO overflow flag Jx (FDOVFJx) of the timer status register Jx (TSRJx) is set to 1, writing 1 to this register clears FDOVFJx to 0. This register is always read as 0.

(4) FDFFCJx — FIFO Table Full Flag Clear Jx Enable

When FIFO data full flag Gx (FDFFJx) of the timer status register Jx (TSRJx) is set to 1, writing 1 to this register clears FDFFJx to 0. By reading data captured in FIFOJx, the data saved in the FIFO register must be made less than the amount specified by the FDFTRGJx bit.

21.12.2.6 TCNTJx — Timer Counter Jx

(x = 0, 1, 2, 3, 4, 5: Corresponding to subblocks J0 to J5)

Access: 32-bit access is possible. 8-bit access and 16-bit access are prohibited.

Address: FFE6 3C24$_H$ + (20$_H$ * x)

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TCNTJx[31:16]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TCNTJx[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.157 TCNTJx Register Contents

Bit Position	Bit Name	Function
31 to 0	TCNTJx	Timer Counter Jx These bits store the up-counter value.

Timer counters Jx (TCNTJx) are 32-bit readable/writable registers. These registers are provided one for each subblock and when the STRJx bit in the TSTRJ register is 1, these registers are incremented by the clock selected from lines 0 to 6 of the clock bus depending on the control register. The counter value is cleared to 0000 0000$_H$ by the edge of pin TIJx. Only when contention between clearing by the edge input of pin TIJx and timing of count-up occurs, TCNTJx is cleared to 0000 0001$_H$.

An overflow interrupt request can be issued to CPU when the timer counter overflows.

When TCNTJx are reset, they are initialized to 0000 0000$_H$.

21.12.2.7 OCRJx — Compare Match Registers Jx

(x = 0, 1, 2, 3, 4, 5: Corresponding to subblocks J0 to J5)

Access: 32-bit access is possible. 8-bit access and 16-bit access are prohibited.

Address: FFE6 $3C28_H$ + (20_H * x)

Value after reset: $FFFF\ FFFF_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	OCRJx[31:16]															
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	OCRJx[15:0]															
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.158 OCRJx Register Contents

Bit Position	Bit Name	Function
31 to 0	OCRJx	Compare Match Jx These bits store the compare match value.

Compare match registers OCRJx are 32-bit readable/writable registers. These registers are provided one for each subblock and function as output compare registers for timer counter Jx (TCNTJx).

When OCRJ0 to OCRJ5 are reset, they are initialized to $FFFF\ FFFF_H$.

When FIFOENJx and FVCRENJx in FIFO control register Jx (FCRJx) is 0 and 1, and STRJx bit in the TSTRJ register is 1, respectively, the FIFO is enabled (FVLDFJx is set to 1) on TCNTJx and OCRJx compare match.

21.12.2.8 FIFOJx — FIFO Registers Jx

(x = 0, 1, 2, 3, 4, 5: Corresponding to subblocks J0 to J5)

Access: 32-bit access is possible. 8-bit access and 16-bit access are prohibited.

Address: FFE6 3C2C$_H$ + (20$_H$ * x)

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	FIFODJx[31:16]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	FIFODJx[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 21.159 FIFOJx Register Contents

Bit Position	Bit Name	Function
31 to 0	FIFODJx	FIFO Data Jn These bits store the FIFO register data.

FIFO registers Jx (FIFOJx) are 32-bit read-only registers. These register are provided one for each subblock and can store nine words of data of timer counter Jx (TCNTJx). FIFOJx can be read but cannot be written by the CPU. If FIFOJx in which no data is captured is read, the read value is undefined.

If FIFO registers hold nine pieces of capture data (when FDNRJx is 9), an occurrence of capture can issue an interrupt request of FIFO data overflow to CPU. At this time, the new captured data is lost. If another capture occurs while nine words of data is captured in the FIFO, the latest data is lost. Secure the free area by reading the FIFO before another capture has occurred (FDNRJx is equal to or less than 9).

21.12.2.9 FDNRJx — FIFO Data Count Registers Jx

(x = 0, 1, 2, 3, 4, 5: Corresponding to subblocks J0 to J5)

Access: 8-bit access is possible.

Address: FFE6 3C30$_H$ + (20$_H$ * x)

Value after reset: 00$_H$

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	FDNJx			
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 21.160 FDNRJx Register Contents

Bit Position	Bit Name	Function
7 to 4	—	Not used. Fix these bits to 0.
3 to 0	FDNJx	FIFO Data Count Jx These bits indicate the number of data words captured in the FIFO. The value ranges from 0$_H$ to 9$_H$. 0$_H$ indicates no data is in the FIFO and 9$_H$ indicates that FIFOJx is full with the captured data.

Note 1. This register is a read-only register. Writing to this register is ignored.

FIFO data count registers Jx (FDNRJx) are 8-bit read-only registers. These registers are provided one for each subblock and indicate the number of data words in FIFOJx which stores captured data. This register cannot be written to.

This register is incremented by 1 every capture in FIFO and is decremented by 1 every read from FIFO (one-data read).

When the FIFO is reset by the FRSTJx bit in FIFO control register Jx (FCRJx), the value in this register is also cleared to 0$_H$.

21.12.2.10 NCNTJx — Noise Canceler Counters Jx

(x = 0, 1, 2, 3, 4, 5: Corresponding to subblocks J0 to J5)

Access: 16-bit access is possible. 8-bit access is prohibited.

Address: FFE6 $3C34_H$ + (20_H * x)

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	NCNTJx															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.161 NCNTJx Register Contents

Bit Position	Bit Name	Function
15 to 0	NCNTJx	Noise Canceler Counter Jx These bits store the 16-bit counter value.

Noise cancel counters Jx (NCNTJx) are 16-bit readable/writable registers.

These registers are incremented by an assertion of the input signals (TIJx) as triggers when the noise canceler function is enabled by the noise canceler enable bits (NCEJx) in timer control register Jx (TCRJx). These registers are driven by the noise canceler counter clock, which is supplied by the prescaler. In level accumulation cancellation mode, up/down counting is performed based on the external input level.

NCNTJx is readable/writable only in 16-bit units. When NCNTJx is reset, it is initialized to 0000_H. One of the three different operation modes, which are premature-transition cancellation mode, minimum time-at-level cancellation mode, and level accumulation cancellation mode, takes place depending on the setting of the noise cancel mode selection bit (NCMSEL) and timer J noise cancel mode bit (NCMF) of the noise cancel mode register (NCMR) of the common controller.

- In premature-transition cancellation mode
  When the NCEJx bit is 1, NCNTJx is stopped, and a level change on pin TIJx is detected, NCNTJx is started for counting up. The counter is cleared to 0000_H and stopped on the first edge of the PCLK after the counter value matches the value in noise canceler register Jx (NCRJx). NCNTJx is incremented regardless of the TJE bit in the ATU-IV master enable register (ATUENR).
  The level change at the start of counting is output as the signal after noise removal and its edge is to be extracted. However, since subsequent level changes are masked, the noise removal signal is not changed. When the values in the counter and NCRJx match, the input signal level at this time is output as the noise removal signal.
  When NCEJx bits are cleared to 0 while the counter is being incremented, counting continues until the values in the counter and the noise canceler register match. The subsequent level changes are also masked over this period.

- In minimum time-at-level cancellation mode
  When the NCEJx bit is 1, NCNTJx is stopped, and a level change on pin TIJx is detected, NCNTJx is started for counting up. If subsequent level change is detected or the values in the counter and the noise canceler register Jx (NCRJx) match, the counter is cleared to 0000_H and stopped on the next PCLK cycle.
  NCNTJx is incremented regardless of the TJE bit in the ATU-IV master enable register (ATUENR).
  The signal after noise removal is changed only on compare match between the counter and NCRJx in synchronization with the level change at the start of counting. When the counter is stopped before the compare match, level changes at the start and stop of counting are masked and the signal after noise removal is not changed.
  When the NCEJx bit is cleared during the counter in operation, counting continues until compare match or a level change on the pin is detected.

- Level accumulation cancellation mode
  When the NCEJx bit is set to 1, NCNTJx performs up or down counting depending on the input signal level. Up-counting is performed when the input level is high, and if the counter value coincides with noise cancel register Jx (NCRJx), the up-counting stops at the next PCLK cycle. Down-counting is performed when the input level is low, and if the counter value coincides with 0000_H, the down-counting stops at the next PCLK cycle.
  Counting of NCNTJx is performed regardless of the setting of the TJE bit of the ATU-IV master enable register (ATUENR).
  If an up-count results in a compare match of NCRJx, the noise canceler output is updated to 1. If a down-count results in a compare match with 0000_H, the noise canceler output is updated to 0.
  If the NCEJx bit is cleared during counting operation, the noise cancel counter stops counting, and the value is changed from the noise canceler output to the current input signal level.
  Be noted that clearing the NCEJx bit in level accumulation cancellation mode might cause an edge detection due to this value change.

21.12.2.11 NCRJx — Noise Cancel Registers Jx

(x = 0, 1, 2, 3, 4, 5: Corresponding to subblocks J0 to J5)

Access: 16-bit access is supported. 8-bit access is prohibited.

Address: FFE6 $3C36_H$ + (20_H * x)

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	NCTJx															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 21.162 NCRJx Register Contents

Bit Position	Bit Name	Function
15 to 0	NCTJx	Noise Cancellation Time Jn These bits store the 16-bit TIJx noise cancellation time.

Noise cancel registers Jx (NCRJx) are 8-bit readable/writable registers that set the upper limitation of the noise cancel counters Jx (NCNTJx). When a period of 128 PCLK cycles is selected as the noise cancel clock, setting of $FFFF_H$ can cancel noise for maximum of 0.21 second period (when PCLK = 40 MHz).

One of the three different operation modes, which are premature-transition cancellation mode, minimum time-at-level cancellation mode, and level accumulation cancellation mode, takes place depending on the setting of the noise cancel mode selection bit (NCMSEL) and timer J noise cancel mode bit (NCMF) of the noise cancel mode register (NCMR) of the common controller.

- Premature-transition cancellation mode
  While NCNTJx counting is in progress, level changes of succeeding input signals are masked. Values of NCNTJx and NCRJx are always compared. If a compare match occurs, the NCNTJx counter value is cleared, the counting is stopped, and the input signal mask is removed at the next PCLK cycle.
- Minimum time-at-level cancellation mode
  While NCNTJx counting is in progress, the noise canceler processing is waited for. Values of NCNTJx and NCRJx are always compared. If a compare match occurs, the NCNTJx counter value is cleared, the counting is stopped, and the noise canceler starts to output noise-canceled signals at the next PCLK cycle.
- Level accumulation cancellation mode
  When up-counting of NCNTJ1 and NCNTJ0 is in progress, values of NCNTJx and NCRJx are compared. When a compare match occurs, the up-counting of NCNTJx stops at the next PCLK cycle. When NCNTJx down-counting is in progress, NCNTJx is compared with 0000_H.

NCRJx is readable/writable only in 16-bit units.

When NCRJx is reset, it is initialized to 0000_H.

21.12.3 Operation description

The TCNTJx counter clock is selected by the TCRJx register from the clock-bus lines. When the TJE bit in ATUENR and the STRJx bit in TSTR is set to 1, TCNTJx can be operated. TCNTJx is cleared to 0000 0000_H by the input edge of pin TIJx. Only when contention between clearing by the edge input of pin TIJx and count-up timing occurs, TCNTJx is cleared to 0000 0001_H.

For the edge detection, the rising, falling, or both edge sensing can be selected by the IOJx bit in TCRJx.

When the FIFOEN bit in FCRFJx is set to 1, FIFOJx captures the TCNTJx counter value by the input edge of pin TIJx. The number of data captured in the FIFO is indicated in FDNRx.

When FIFOJx has taken data in a quantity equal to or larger than the number of data set in the FDFTRGJx bit of FCRJx, a FIFO data full interrupt request to CPU and a DMA transfer request to DMAC can be issued. FDFFJx of TSRJx is set to 1.

When FIFOENJx = 0, FVCRENJx = 1, and compare match between TCNTJx and OCRJn occurs, FIFOJx is enabled (FVLDFJx in TSRJx is 1). Under these conditions, when the number of data captured in FIFOJx exceeds the value set in the FDFTRGJx bits in FCRJx, FIFOJx is disabled.

Operation examples for subblock 0 are shown below. The TCNTJ0 counter clock is an ideal signal to show the counting and clearing timing of TCNTJ0.

In **Figure 21.75**, falling edge sensing is selected, FIFOEN = 1, FDFTRGJx = 11_B, and FDFFJx is set by the input edge.

In **Figure 21.76**, falling edge sensing is selected, FIFOEN = 0, FVCRENJx = 1, FDFTRGJx = 00_B and FDFFx is set by the input edge.

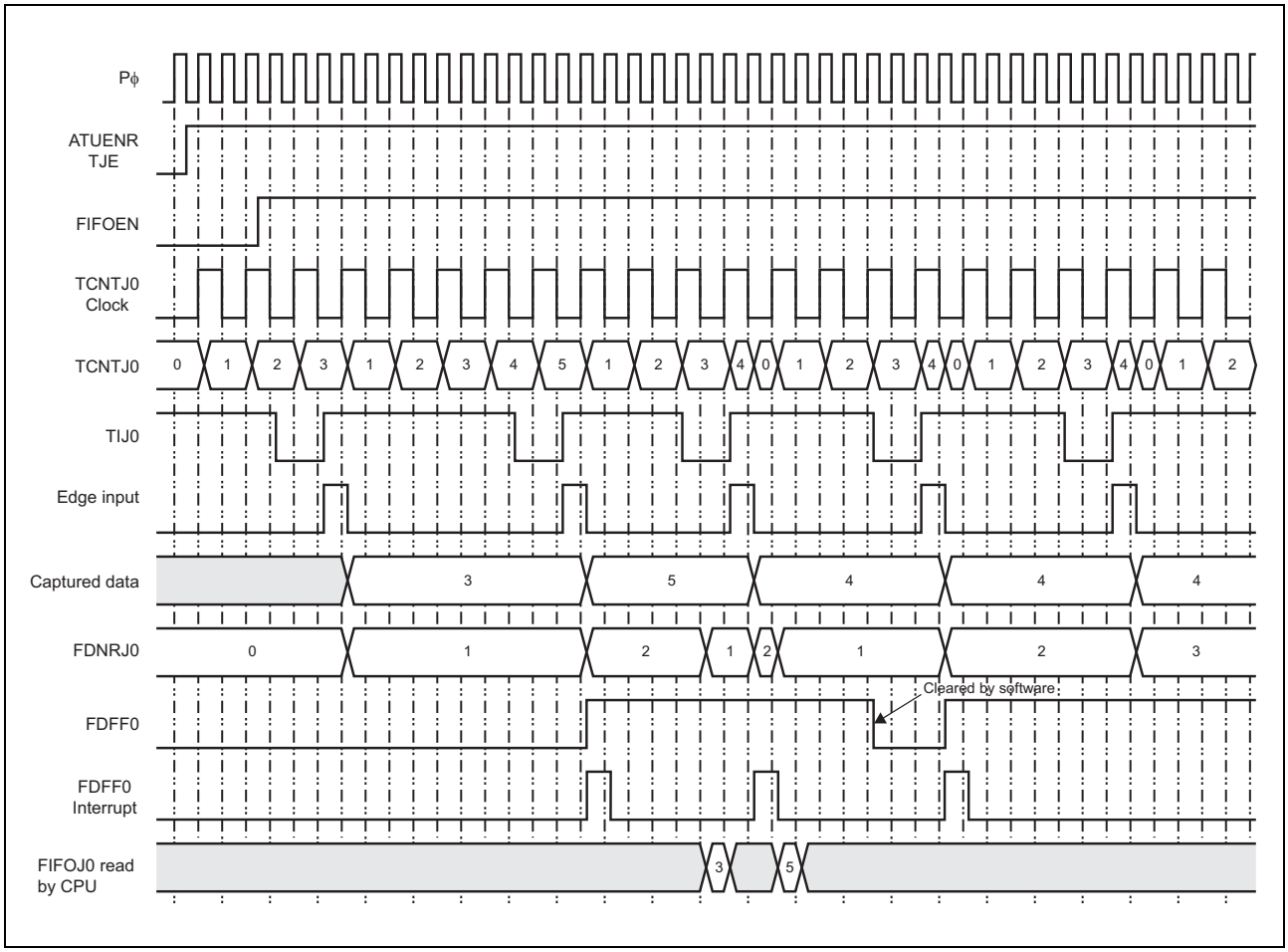


Figure 21.75 Operation Example of Timer J (1)

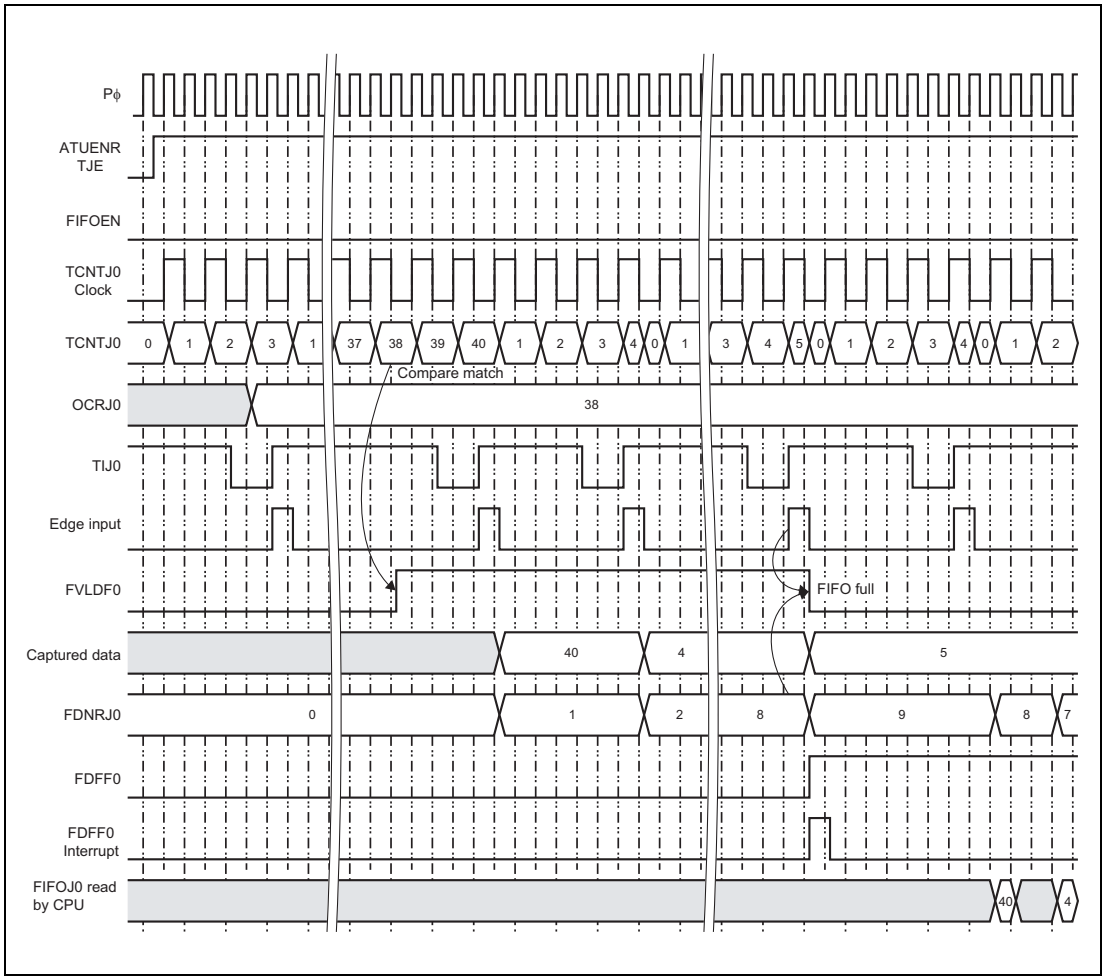


Figure 21.76 Operation Example of Timer J (2)

21.13 Automatic Switching of DMA and AD Requests

21.13.1 Overview of Operation

This function controls trigger signals to be output to DMA and SAR-AD. There are three trigger signals (data DMA trigger, count DMA trigger, and SAR-AD trigger signals) described below.

(1) DMA trigger for data — Data DMA trigger

Data DMA trigger signals include AD skipping DMA transfer request 0 and AD skipping DMA transfer request 1.

The data DMA trigger can be selected from AGCK1 from timer B, timer G sub-block G1, G2 compare interrupt output (timer G1 compare interrupt or timer G2 compare interrupt), DFE filter completion signal, ADC0 scan group 0 to 4 end interrupt (ADC0 scan group 0 end interrupt, ADC0 scan group 1 end interrupt, ADC0 scan group 2 end interrupt, ADC0 scan group 3 end interrupt, or ADC0 scan group 4 end interrupt), and output stop. One of the two data DMA trigger signals is output, and the DMA output to output the trigger is switched at the comparison timing of TCNT1D0 and OCR1D00 of timer D.

(2) DMA trigger for the number of times — Count DMA trigger

Count DMA trigger signals include AD skipping DMA transfer request 2 and AD skipping DMA transfer request 3.

The count DMA trigger can be selected from AGCK of timer A, timer G sub-block G3 compare interrupt output (timer G3 compare interrupt), and output stop. One of the two count DMA trigger signals is output, and the DMA output to output the trigger is switched at the comparison timing of TCNT1D0 and OCR1D00 of timer D.

(3) SAR-AD trigger

SAR-AD trigger signals include the SAR-AD trigger.

The SAR-AD trigger can be selected from AGCK1 from timer B, timer G sub-block G1, G2 compare interrupt output (timer G1 compare interrupt or timer G2 compare interrupt), and output stop.

21.13.2 DMA/AD Requests Auto-Switching Registers

21.13.2.1 TRGSRDMA0 — Trigger Status Register DMA0

Access: 8-bit accessible, but 16-bit/32-bit inaccessible

Address: FFE6 $00C0_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	OUTSRDMA0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 21.163 TRGSRDMA0 Register Contents

Bit Position	Bit Name	Function
7 to 1	—	Not used
0	OUTSRDMA0	0: AD skipping DMA transfer request 0 is selected 1: AD skipping DMA transfer request 1 is selected

Trigger status register DMA0 is an 8-bit read-only register that indicates the selected request: data AD skipping DMA transfer request 0 or AD skipping DMA transfer request 1.

21.13.2.2 TRGSELDMA00 — Trigger Select Register DMA00

Access: 8-bit accessible, but 16-bit/32-bit inaccessible

Address: FFE6 00C4$_H$

Value after reset: 00$_H$

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	INSELDMA00			
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 21.164 TRGSELDMA00 Register Contents

Bit Position	Bit Name	Function
7 to 4	—	Not used
3 to 0	INSELDMA00	0000: Output stop (output is fixed to 0) 0001: AGCK1 is selected. 0010: Timer G sub-block G1 compare interrupt signal (CMFG1 interrupt) is selected. 0011: Timer G sub-block G2 compare interrupt signal (CMFG2 interrupt) is selected. 0100: DFE filter completion signal is selected. 0101: Setting prohibited 0110: Setting prohibited 0111: Setting prohibited 1000: ADC0 scan group 0 end interrupt is selected. 1001: ADC0 scan group 1 end interrupt is selected. 1010: ADC0 scan group 2 end interrupt is selected. 1011: ADC0 scan group 3 end interrupt is selected. 1100: ADC0 scan group 4 end interrupt is selected. 1101: Setting prohibited 1110: Setting prohibited 1111: Setting prohibited

This register is an 8-bit readable/writable register that selects the signal to be output to the data AD skipping DMA transfer request 0. This register can be modified during operation. The settings are valid immediately after this register is modified.

21.13.2.3 TRGSELDMA01 — Trigger Select Register DMA01

Access: 8-bit accessible, but 16-bit/32-bit inaccessible

Address: FFE6 00C6$_H$

Value after reset: 00$_H$

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	INSELDMA01			
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 21.165 TRGSELDMA01 Register Contents

Bit Position	Bit Name	Function
7 to 4	—	Not used
3 to 0	INSELDMA01	0000: Output stop (output is fixed to 0) 0001: AGCK1 is selected. 0010: Timer G sub-block G1 compare interrupt signal (CMFG1 interrupt) is selected. 0011: Timer G sub-block G2 compare interrupt signal (CMFG2 interrupt) is selected. 0100: DFE filter completion signal is selected. 0101: Setting prohibited 0110: Setting prohibited 0111: Setting prohibited 1000: ADC0 scan group 0 end interrupt is selected. 1001: ADC0 scan group 1 end interrupt is selected. 1010: ADC0 scan group 2 end interrupt is selected. 1011: ADC0 scan group 3 end interrupt is selected. 1100: ADC0 scan group 4 end interrupt is selected. 1101: Setting prohibited 1110: Setting prohibited 1111: Setting prohibited

This register is an 8-bit readable/writable register that selects the signal to be output to the data AD skipping DMA transfer request 1. This register can be modified during operation. The settings are valid immediately after this register is modified.

21.13.2.4 TRGSELAD — Trigger Select Register AD

Access: 8-bit accessible, but 16-bit/32-bit inaccessible

Address: FFE6 00C8$_H$

Value after reset: 00$_H$

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	INSELAD	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 21.166 TRGSELAD Register Contents

Bit Position	Bit Name	Function
7 to 2	—	Not used
1, 0	INSELAD	00: Output stop (output is fixed to 0) 01: AGCK1 is selected. 10: Timer G sub-block G1 compare interrupt signal (CMFG1 interrupt) is selected. 11: Timer G sub-block G2 compare interrupt signal (CMFG2 interrupt) is selected.

This register is an 8-bit readable/writable register that selects the signal to be output to the SAR-AD trigger. This register can be modified during operation. The settings are valid immediately after this register is modified.

21.13.2.5 TRGSRDMA1 — Trigger Status Register DMA1

Access: 8-bit accessible, but 16-bit/32-bit inaccessible

Address: FFE6 00D0$_H$

Value after reset: 00$_H$

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	OUTSRDMA1
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 21.167 TRGSRDMA1 Register Contents

Bit Position	Bit Name	Function
7 to 1	—	Not used
0	OUTSRDMA1	0: AD skipping DMA transfer request 2 is selected. 1: AD skipping DMA transfer request 3 is selected.

This register is an 8-bit read-only register that indicates the selected request (count AD skipping DMA transfer request 2 or AD skipping DMA transfer request 3).

21.13.2.6 TRGSELDMA10 — Trigger Select Register DMA10

Access: 8-bit accessible, but 16-bit/32-bit inaccessible

Address: FFE6 00D4$_H$

Value after reset: 00$_H$

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	INSELDMA10	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 21.168 TRGSELDMA10 Register Contents

Bit Position	Bit Name	Function
7 to 2	—	Not used
1, 0	INSELDMA10	00: Output stop (output is fixed to 0) 01: AGCK is selected. 10: Timer G sub-block G3 compare interrupt signal (CMFG3 interrupt) is selected. 11: Setting prohibited

This register is an 8-bit readable/writable register that selects the signal to be output to the count AD skipping DMA transfer request 2. This register can be modified during operation. The settings are valid immediately after this register is modified.

21.13.2.7 TRGSELDMA11 — Trigger Select Register DMA11

Access: 8-bit accessible, but 16-bit/32-bit inaccessible

Address: FFE6 00D6$_H$

Value after reset: 00$_H$

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	INSELDMA11	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 21.169 TRGSELDMA11 Register Contents

Bit Position	Bit Name	Function
7 to 2	—	Not used
1, 0	INSELDMA11	00: Output stop (output is fixed to 0) 01: AGCK is selected. 10: Timer G sub-block G3 compare interrupt signal (CMFG3 interrupt) is selected. 11: Setting prohibited

This register is an 8-bit readable/writable register that selects the signal to be output to the count AD skipping DMA transfer request 3. This register can be modified during operation. The settings are valid immediately after this register is modified.

21.13.3 Details of Operation

The following describes the operation and block diagram of each trigger.

21.13.3.1 Data DMA Trigger

Figure 21.77 shows the data DMA trigger selection circuit.

The signal to be output to the data DMA trigger output pin (AD skipping DMA transfer request 0) is selected by the INSELDMA00[3:0] bits in TRGSELDMA00.

The signal to be output to the data DMA trigger output pin (AD skipping DMA transfer request 1) is selected by the INSELDMA01[3:0] bits in TRGSELDMA01.

The data DMA trigger output pin outputs only AD skipping DMA transfer request 0 when the DMA select signal is low, and only AD skipping DMA transfer request 1 when the DMA select signal is high. Since ADC0 scan group 0 end interrupt to ADC0 scan group 4 end interrupt are asynchronous input signals, a synchronization circuit is provided.

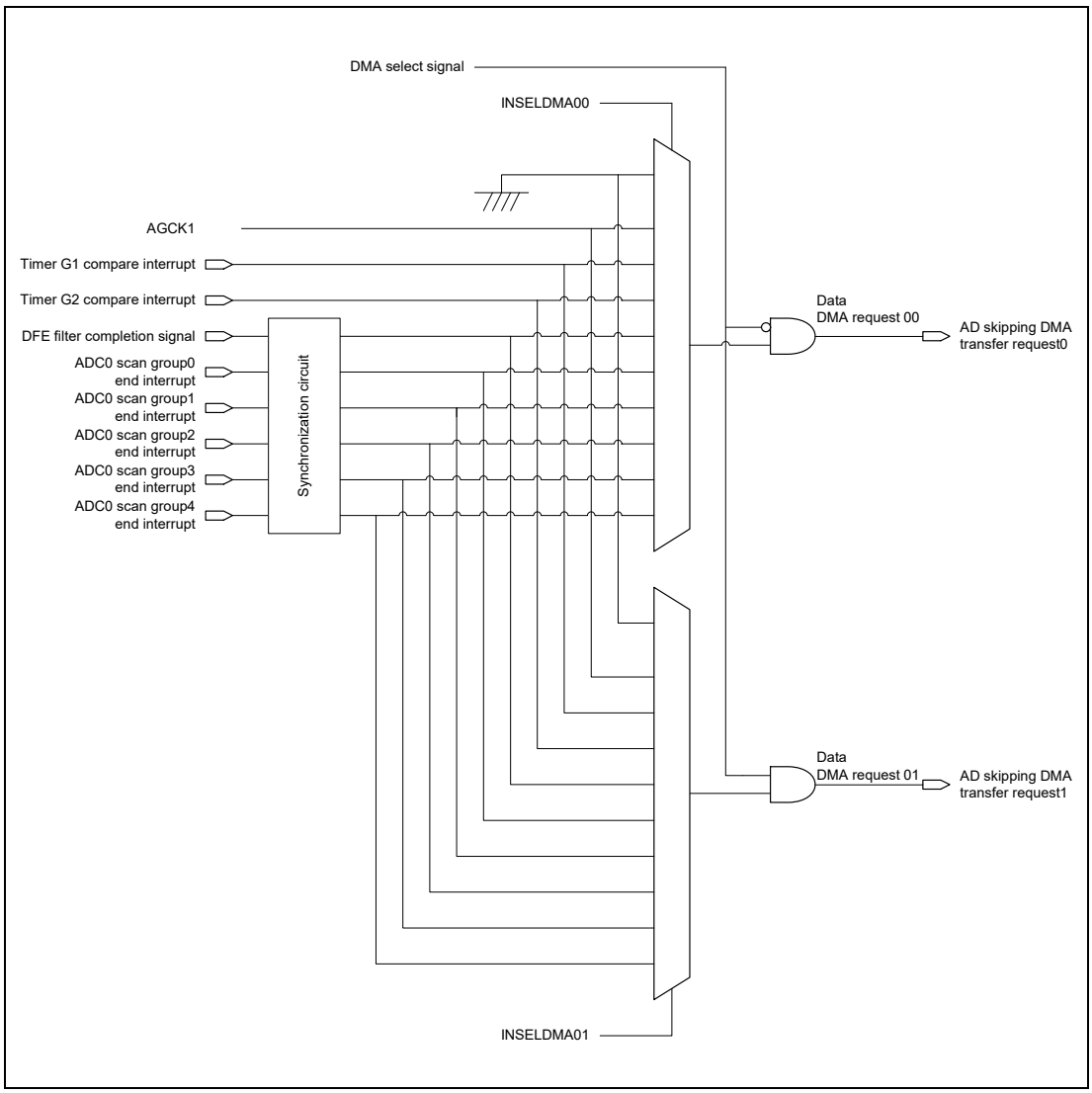


Figure 21.77 Data DMA Trigger Selection Circuit

21.13.3.2 Count DMA Trigger

Figure 21.78 shows the count DMA trigger selection circuit.

The signal to be output to the count DMA trigger output pin (AD skipping DMA transfer request 2) is selected by the INSELDMA10[1:0] bits in TRGSELDMA10.

The signal to be output to the count DMA trigger output pin (AD skipping DMA transfer request 3) is selected by the INSELDMA11[1:0] bits in TRGSELDMA11.

The count DMA trigger output pin outputs only AD skipping DMA transfer request 2 when the DMA select signal is low, and only AD skipping DMA transfer request 3 when the DMA select signal is high.

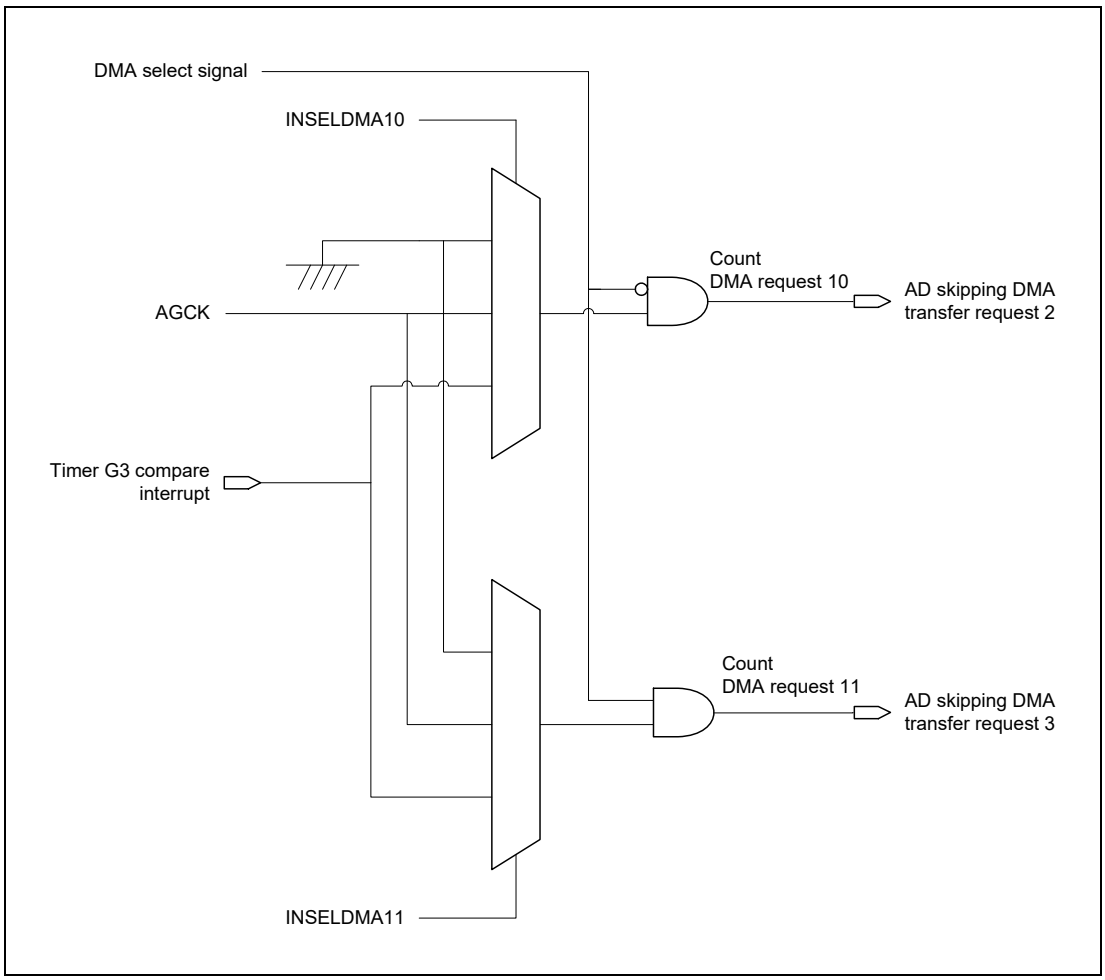


Figure 21.78 Count DMA Trigger Selection Circuit

21.13.3.3 SAR-AD Trigger

Figure 21.79 shows the SAR-AD trigger selection circuit.

The signal to be output to the SAR-AD trigger output pin is selected by the INSELAD[1:0] bits in TRGSELAD.

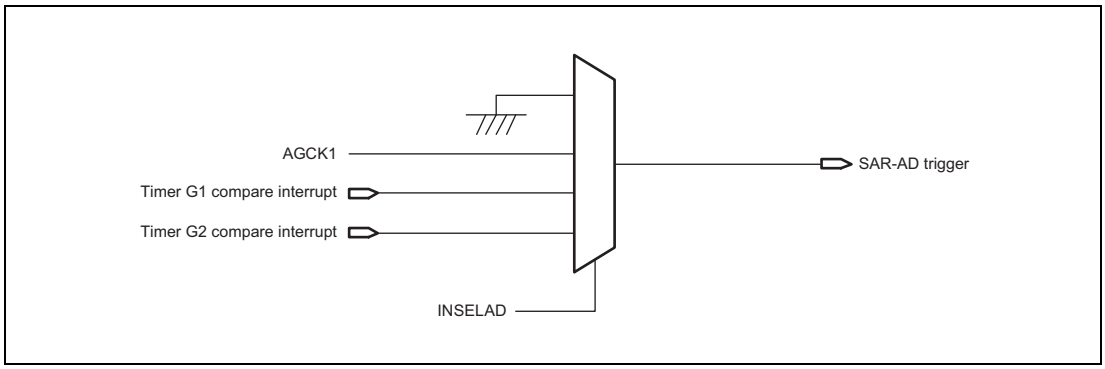


Figure 21.79 SAR-AD Trigger Selection Circuit

Figure 21.80 shows the trigger output timing chart.

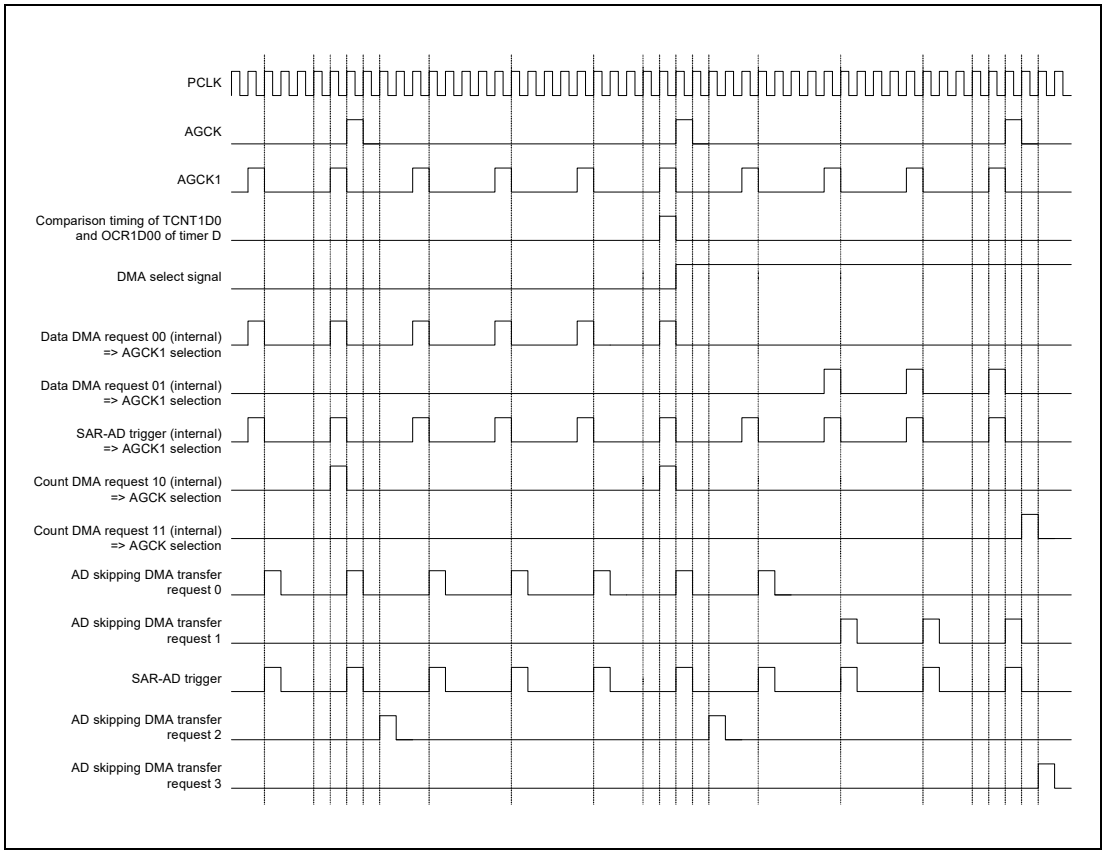


Figure 21.80 DMA/AD Trigger Timing Chart

21.14 Usage Notes

In ATU-IV, status register (TSR) flags deployed in different timers are status flags indicating occurrences of compare match, overflow, underflow, and input capture. In the condition that the status flag is set to 1, an occurrence of compare match, overflow, underflow, or input capture retains the status flag value of 1, and issues an interrupt request to CPU. To find out whether an interrupt request has been made properly, the corresponding flag on the “INTC” side needs to be checked. Be noted that if two interrupt conditions occur in succession in a period shorter than a PCLK cycle, an interrupt request is generated for the first interrupt condition, but not for the subsequent one.

Note that contention and operation described below occur during ATU-IV operation.

21.14.1 Input Capture Contention

21.14.1.1 Contention between Writing to General Register and Input Capture

When writing to a general register occurs simultaneously with input capture, writing takes priority (**Figure 21.81**). However, an input capture interrupt request is issued, and if input capture status is provided, the input capture flag is set. The waveforms in the right half of **Figure 21.81** indicate a case in which writing occurs one PCLK cycle prior to input capture.

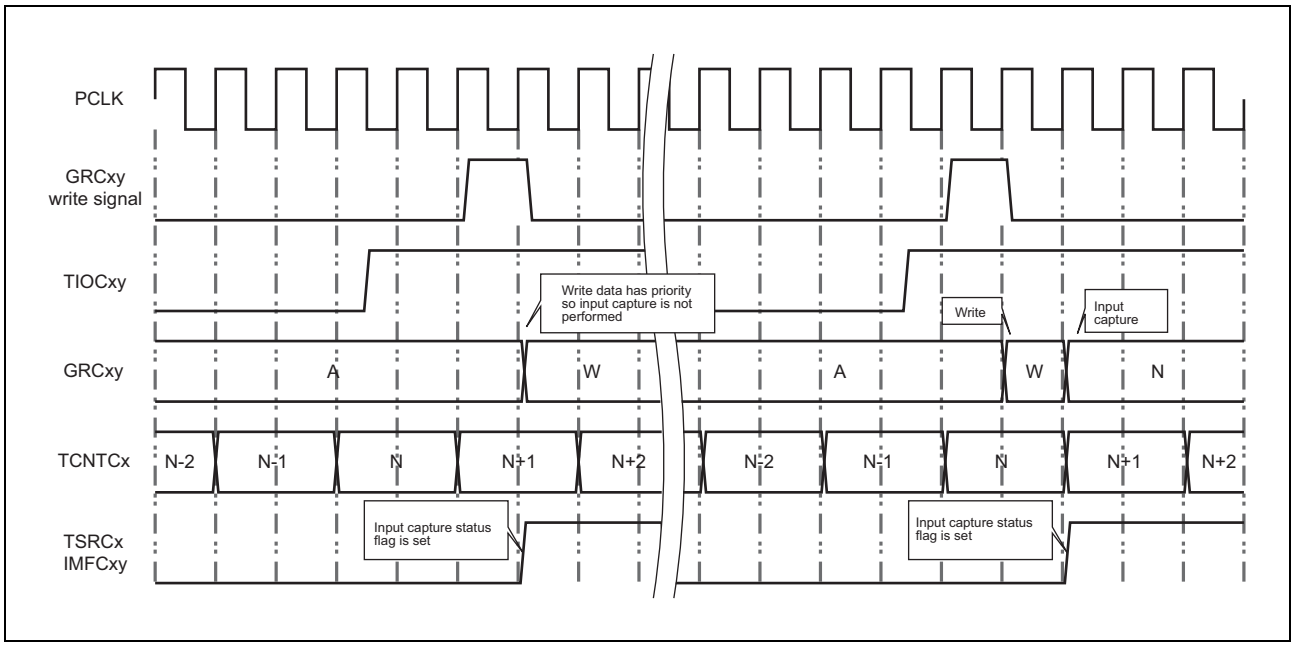


Figure 21.81 Contention between Writing to GRCxy and Input Capture

Timer	Counter (Whose Value is Captured)	Capture Register	Status
Timer C	TCNTCx	GRCxy	IMFCxy
Timer F	ECNTAFx	GRAFx	ICFFx
	ECNTBFx	GRBFx	
	ECNTCFx	GRCFx	
	ECNTCFx + GRDFx	GRDFx	

21.14.1.2 Contention between Writing to Counter and Input Capture

When writing to a counter occurs simultaneously with input capture, the value immediately before writing is captured (**Figure 21.82**). The waveforms in the right half of **Figure 21.82** indicate a case in which writing occurs one PCLK cycle prior to input capture, so the written value is captured.

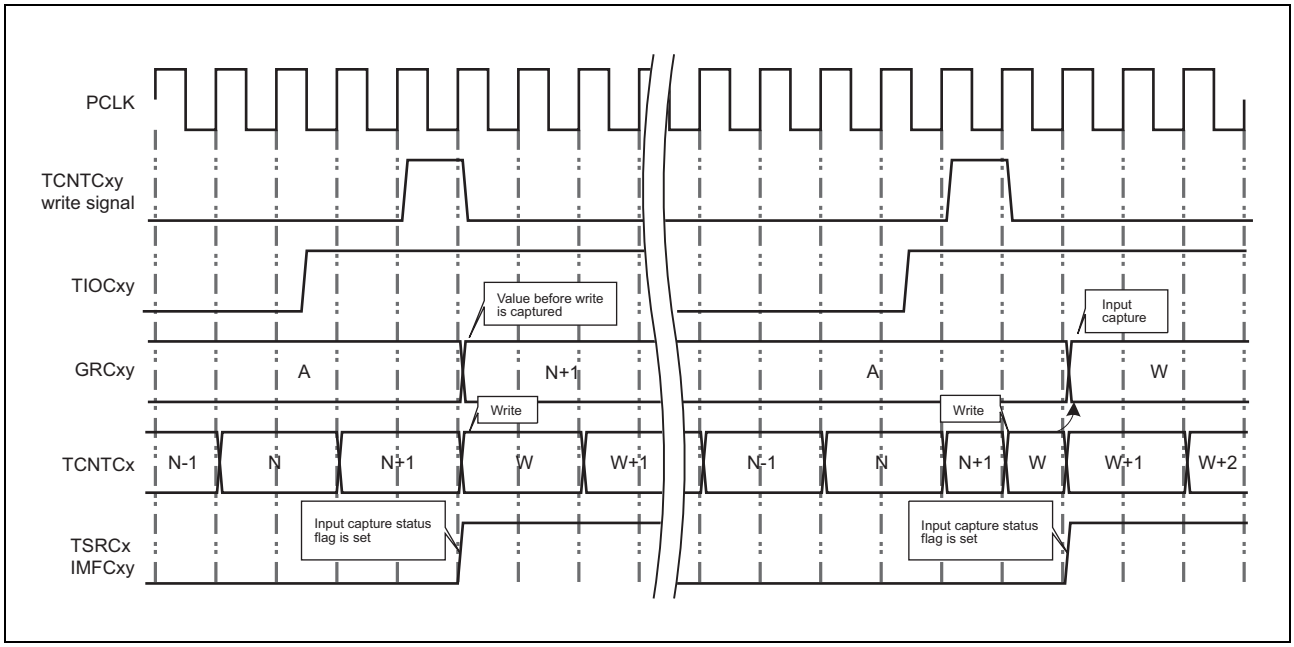


Figure 21.82 Contention between Writing to TCNTCx and Input Capture

Target Timer	Counter (Whose Value is Captured)	Capture Register	Status
Timer A	TCNTAx	ICRAx	ICFAx
Timer B	TCNTB0	ICRB0	ICFB0
	TCNTB0 + ICRB1	ICRB1	
	TCNTB6	ICRB6	
Timer C	TCNTCx	GRCxy	IMFCxy
Timer D	TCNT1Dx	OSBRDx	
	TCNT1Dx	ICR1Dxy	
	TCNT2Dx	ICR2Dxy	
Timer F	ECNTAFx	GRAFx	ICFFx
	ECNTBFx	GRBFx	
	ECNTCFx	GRCFx	
	ECNTCFx + GRDFx	GRDFx	
Timer J	TCNTJx	FIFOJx	(FDOVFJx, FDFFJx)

21.14.1.3 Contention between Setting and Clearing of Input Capture Status Flag

Flag clearing by a timer status clear register takes priority over flag setting by an input capture. The left half of **Figure 21.83** shows an example where an input capture and a flag clearing occur simultaneously, and the flag is cleared. In contrast with this, the waveforms in the right half of **Figure 21.83** indicate an example in which input capture is performed immediately after the flag has been cleared.

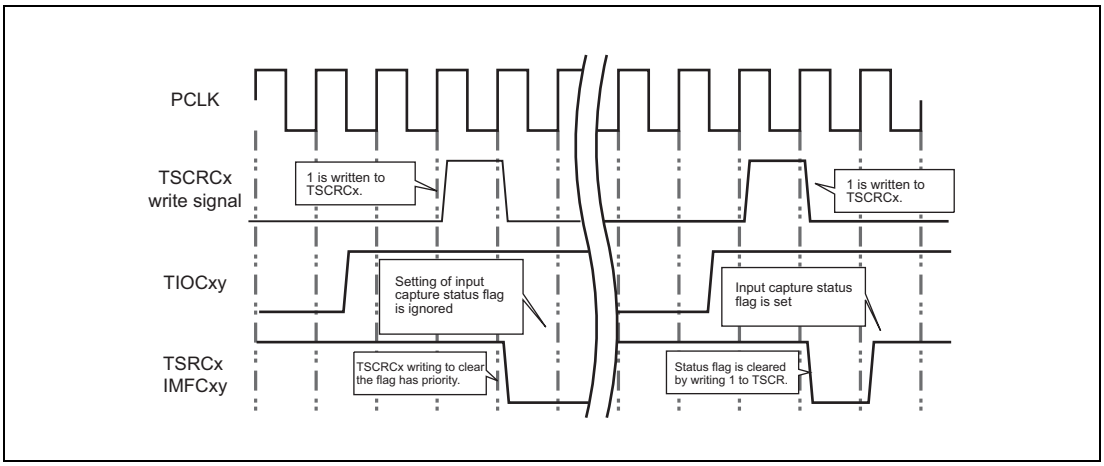


Figure 21.83 Contention between Status Clearing by Writing 1 to Status Clear Register and Input Capture

	Flag
Timer A	ICFAx
Timer B	ICFB0
Timer C	IMFCxy
Timer F	ICFFx
Timer J	(FDOVFJx, FDFFJx)

21.14.2 Compare Match Contention

21.14.2.1 Contention between Writing to Compare-Match General Register and Compare Match

Contention between writing to a register provided with the compare match function and compare match is described here with timer C used as an example. If writing takes place after the values of GRCxy and TCNTCx coincide (left half of **Figure 21.84**), a compare match interrupt request is issued, and the compare match status is set. If GRCxy and TCNTCx do not coincide in any cycles (right half of **Figure 21.84**), no compare match is detected.

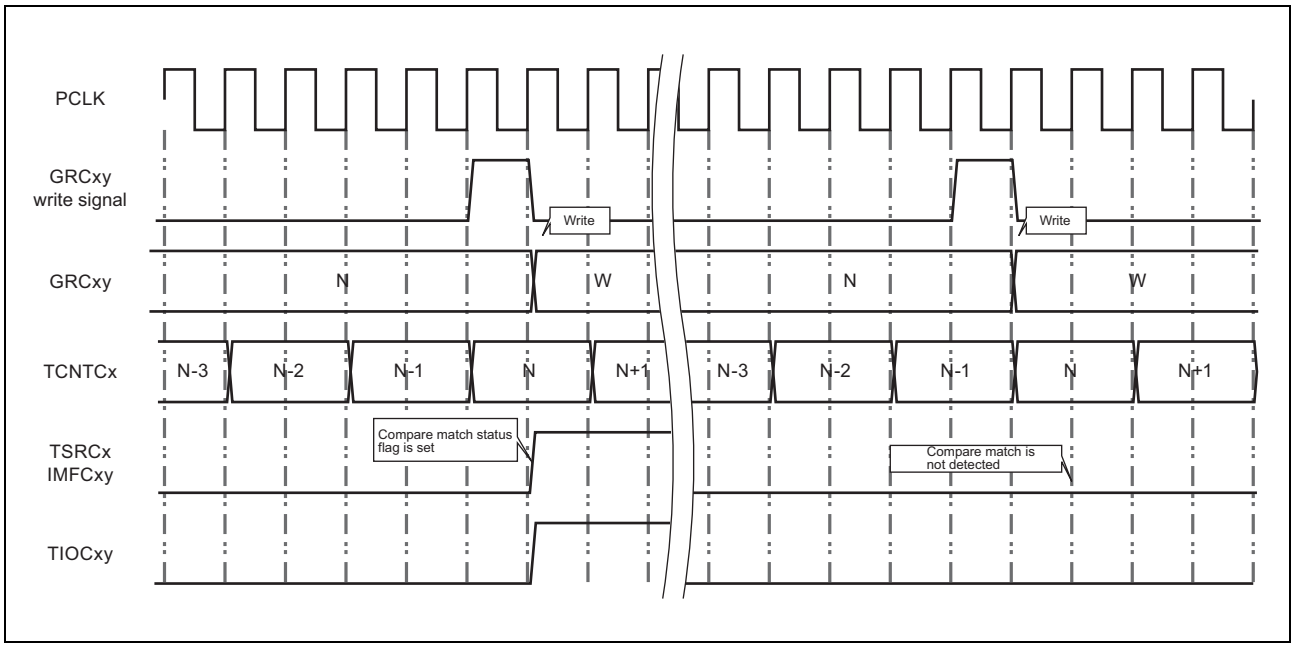


Figure 21.84 Contention between Writing to GRCxy and Compare Match

Timer	Counter	Compare Match Register	Status
Timer B	TCNTB1	OCRB1	CMFB1
	TCNTB1	OCRB10	CMFB10
	TCNTB1	OCRB11	CMFB11
	TCNTB1	OCRB12	CMFB12
Timer C	TCNTCx	GRCxy	IMFCxy
Timer D	TCNT1Dx	OCR1Dxy	CMFADxy
	TCNT2Dx	OCR2Dxy	CMFBDxy
Timer F	ECNTAFx	GRAFx	—
	ECNTBFx	GRBFx	—
	ECNTCFx	GRBFx	OVFCFx
		GRDFx	OVFCFx
Timer G	TCNTGx	OCRGx	CMFGx
Timer H	TCNT1H	OCR1H	CMFH
Timer J	TCNTJx	OCRJx	CMFJx

Timing of compare match B0 and compare match B6 of timer B and the cycle match of timer E differ from the timing of these compare matches. For details, see **Section 21.14.2.2, Contention between Writing to CYLRExy and Cycle Match of TCNTExy**.

21.14.2.2 Contention between Writing to CYLRExy and Cycle Match of TCNTExy

Operation when writing to CYLRExy occurs simultaneously with compare match (cycle match) with TCNTExy is shown below. As shown in **Figure 21.85**, if writing to CYLRExy takes place at the time of cycle match counter clearing, TCNTExy is cleared as in the case of normal cycle match, a cycle match interrupt request is issued, and the cycle match status and PWM output are changed. The waveforms in the right half of **Figure 21.85** show an example in which CYLRExy is written to before the counter is cleared. In this case, no cycle match is detected and TCNTExy continues to be incremented.

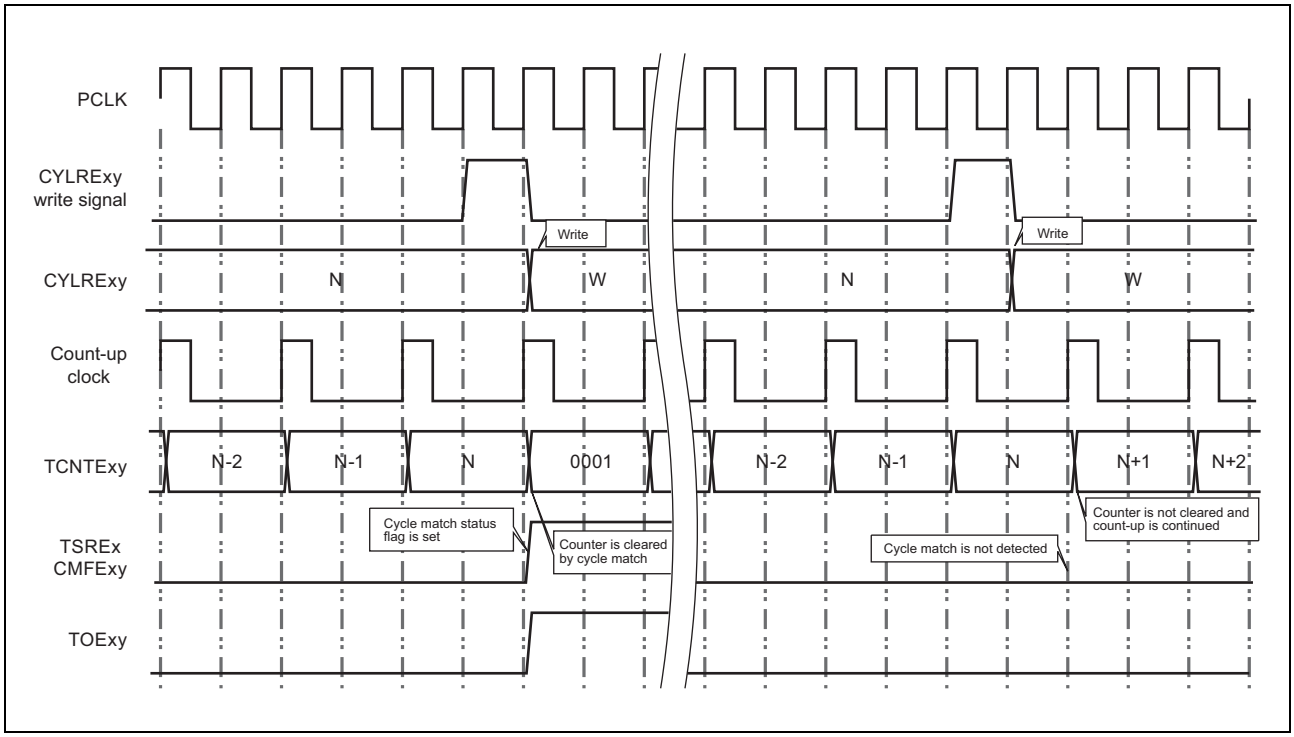


Figure 21.85 Contention between Writing to CYLRExy and Cycle Match

Timer	Counter	Compare (Cycle) Match Register	Status
Timer B	TCNTB0	OCRB0	CMFB0
	TCNTB6	OCRB6/OCRB7	CMFB6
	TCNTB6M	ICRB6	CMFB6M
Timer E	TCNTExy	CYLRExy	CMFExy

21.14.2.3 Contention between Writing to DTRExy and Cycle Match of TCNTExy

The operation is the same as the case of contention between writing to CYLRExy and cycle matches with TCNTExy described above.

21.14.2.4 Contention between Writing to Counter and Compare Match

Contention between writing to a counter and compare match is described below. If writing takes place after the compare match register and the counter coincide (left half of **Figure 21.86**), a compare match interrupt request is issued, and the compare match status is set. If the compare match register and the counter do not coincide in any cycles (right half of **Figure 21.86**), a compare match interrupt request is not is issued, and the match in comparison is not detected.

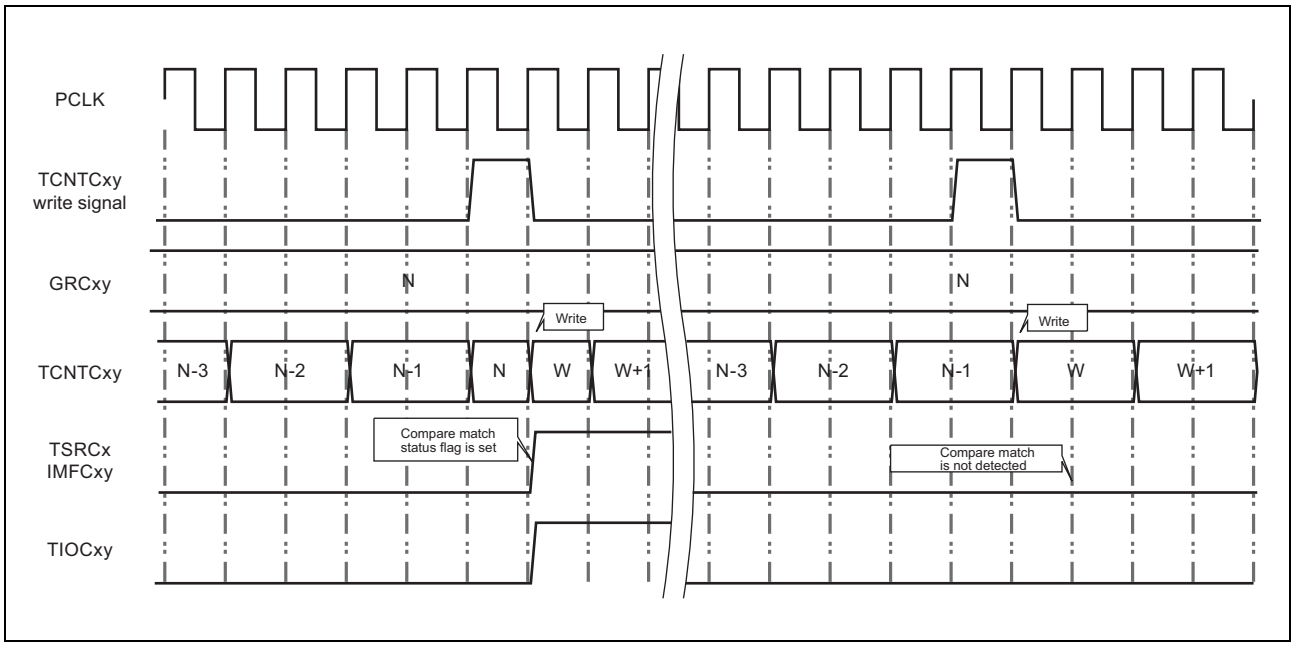


Figure 21.86 Contention between Writing to TCNTCx and Compare Match

Timer	Counter	Compare Match Register	Status
Timer B	TCNTB1	OCRB1	CMFB1
Timer C	TCNTCx	GRCxy	IMFCxy
Timer D	TCNT1Dx	OCR1Dxy	CMFADxy
	TCNT2Dx	OCR2Dxy	CMFBDxy
Timer F	ECNTAFx	GRAFx	—
	ECNTBFx	GRBFx	—
	ECNTCFx	GRBFx	OVFCFx
		GRDFx	OVFCFx
Timer G	TCNTGx	OCRGx	CMFGx
Timer H	TCNT1H	OCR1H	CMFH
Timer J	TCNTJx	OCRJx	CMFJx

Timing of compare match B0 and compare match B6 of timer B and the cycle match of timer E differ from the timing of these compare matches. For details, see **Section 21.14.2.6, Contention between Writing to TCNTExy and Counter Clearing by Cycle Match**.

21.14.2.5 Contention between Writing to Counter and Counter Clearing by Compare Match

The waveforms shown here are for when the function to clear a counter by compare match is enabled. If writing to a counter and counter clearing by compare match occur simultaneously, the counter is not cleared and the write takes precedence (left half of **Figure 21.87**). However, a compare match interrupt request is issued, and the compare match status is set. The waveforms in the right half of **Figure 21.87** show a case in which writing to TCNTCx is one PCLK cycle later.

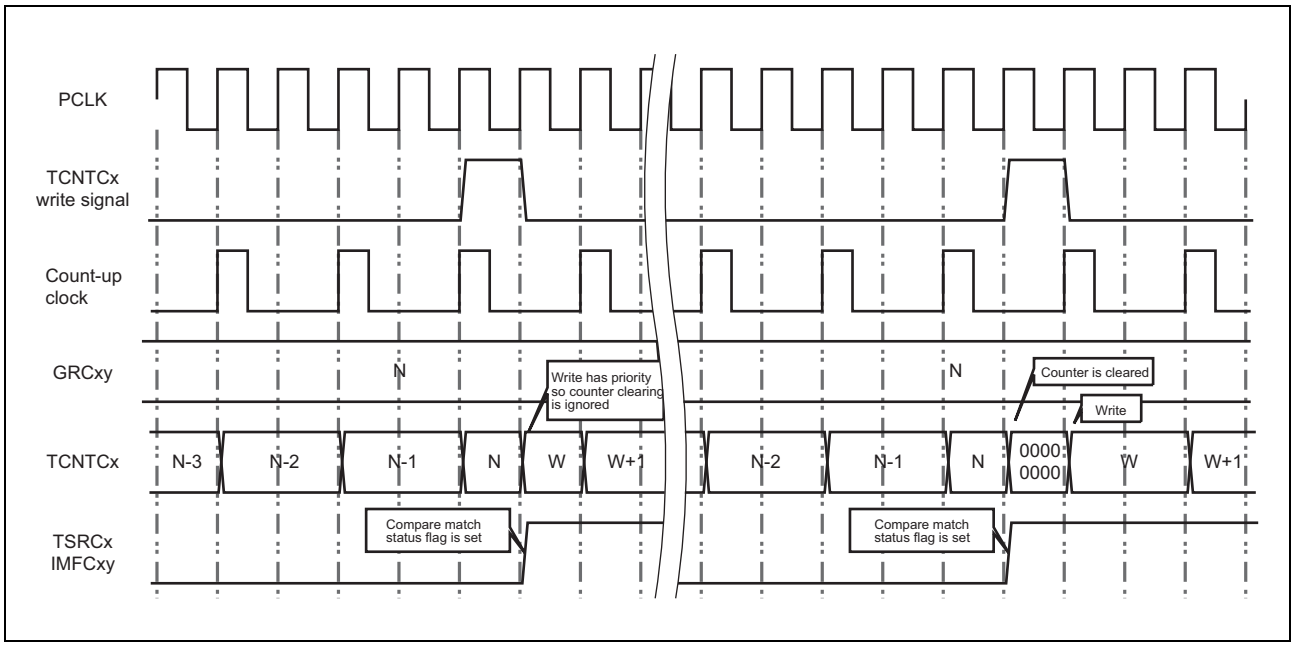


Figure 21.87 Contention between Writing to TCNTCx and Counter Clearing by Compare Match

Timer	Counter	Compare Match Register	Status
Timer C	TCNTCx	GRCxy	IMFCxy
	TCNTCx	CUCRCx	OVFCx
Timer D	TCNT1Dx	CUCR1Dx	OVF1Dx
	TCNT2Dx	CUCR2Dx	OVF2Dx
Timer F	ECNTAFx	GRAFx	—
	ECNTBFx	GRBFx	—
Timer G	TCNTGx	OCRGx	CMFGx
Timer H	TCNT1H	OCR1H	CMFH

21.14.2.6 Contention between Writing to TCNTExy and Counter Clearing by Cycle Match

When writing to TCNTExy occurs simultaneously with counter clearing by cycle match, the counter is not cleared and TCNTExy is written to. However, a cycle match interrupt request is issued, the cycle match status is set, and reload of the cycle-setting register and the duty cycle setting register are performed (right half of **Figure 21.88**). PWM waveform output at the time of cycle match is also performed.

The waveforms in the left half of **Figure 21.88** show operation when writing is performed one PCLK cycle earlier than the count-up clock.

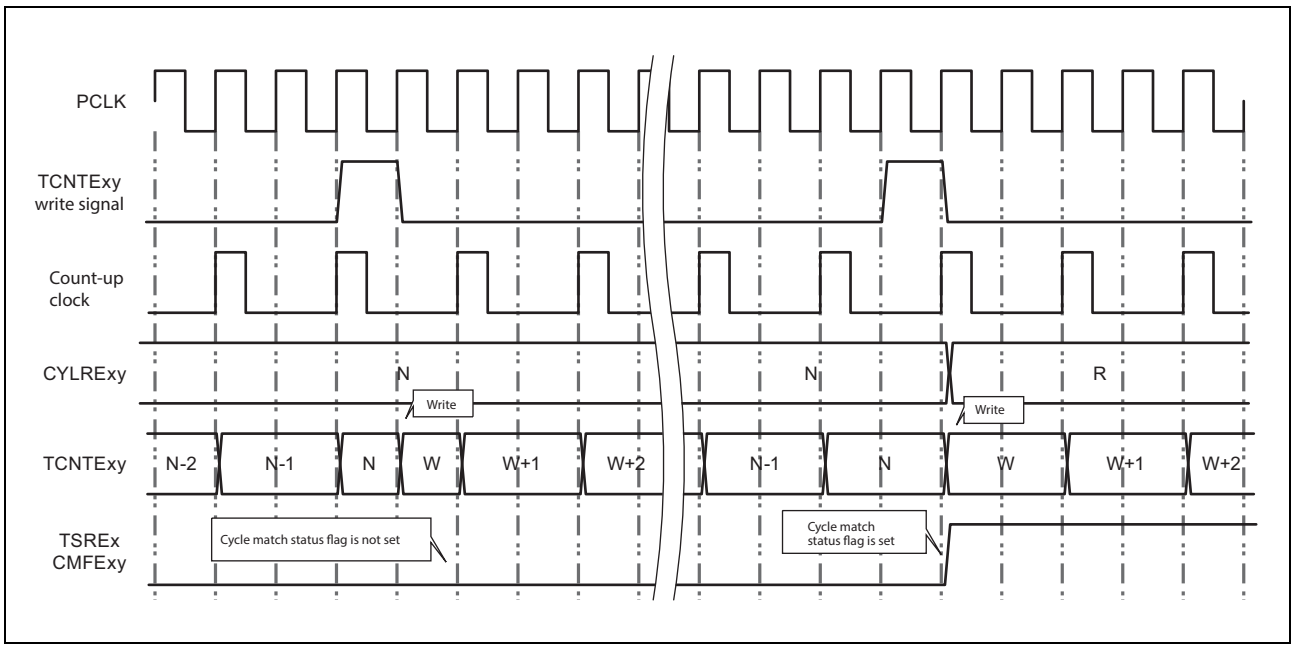


Figure 21.88 Contention between Writing to TCNTExy and Counter Clearing by Cycle Match

Timer	Counter	Compare (Cycle) Match Register	Status
Timer B	TCNTB0	OCRB0	CMFB0
	TCNTB6	OCRB6/OCRB7	CMFB6
	TCNTB6M	ICRB6	CMFB6M
Timer E	TCNTExy	CYLRExy	CMFExy

21.14.2.7 Contention between Setting and Clearing of Compare Match Status Flag

Flag clearing by a timer status clear register takes priority over flag setting by compare match (left half of **Figure 21.89**). The right half of **Figure 21.89** illustrates the case where a compare match occurs just after flag clearing.

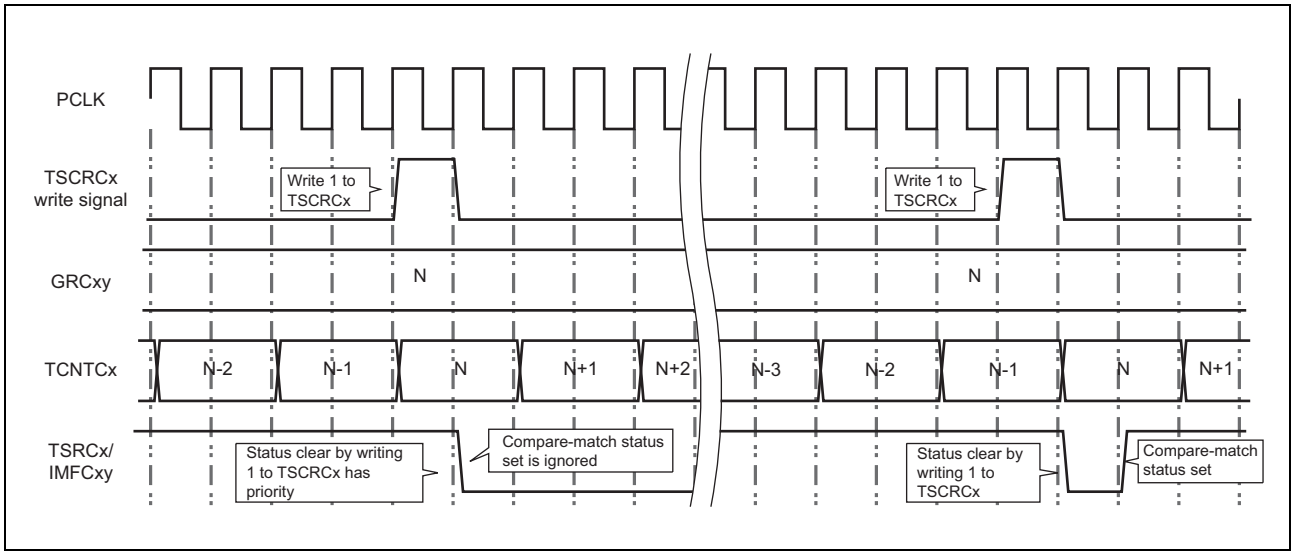


Figure 21.89 Contention between Setting and Clearing of Compare Match Status Flag

Timer	Counter	Compare Match Register	Status
Timer B	TCNTB1	OCRB1	CMFB1
	TCNTB1	OCRB10	CMFB10
	TCNTB1	OCRB11	CMFB11
	TCNTB1	OCRB12	CMFB12
Timer C	TCNTCx	GRCxy	IMFCxy
Timer D	TCNT1Dx	OCR1Dxy	CMFADxy
	TCNT2Dx	OCR2Dxy	CMFBDxy
Timer F	TCNTCFx	GRDFx	OVFCFx
Timer G	TCNTGx	OCRGx	CMFGx
Timer H	TCNT1H	OCR1H	CMFH
Timer J	TCNTJx	OCRJx	CMFJx

Timing of compare match B0 and compare match B6 of timer B and the cycle match of timer E differ from the timing of these compare matches. For details, see **Section 21.14.2.8, Contention between Setting of Cycle Match Status Flag and Writing 1 to the Status Clear Register**.

21.14.2.8 Contention between Setting of Cycle Match Status Flag and Writing 1 to the Status Clear Register

If setting of the cycle match flag (cycle match) and clearing of the flag by the timer status clear register occurs simultaneously, clearing of the flag takes priority. The left half of **Figure 21.90** illustrates an example where flag setting by cycle match and flag clearing by the timer status clear register occur simultaneously. The waveforms in the right half of **Figure 21.90** show an example in which the flag is cleared one PCLK cycle earlier than flag setting.

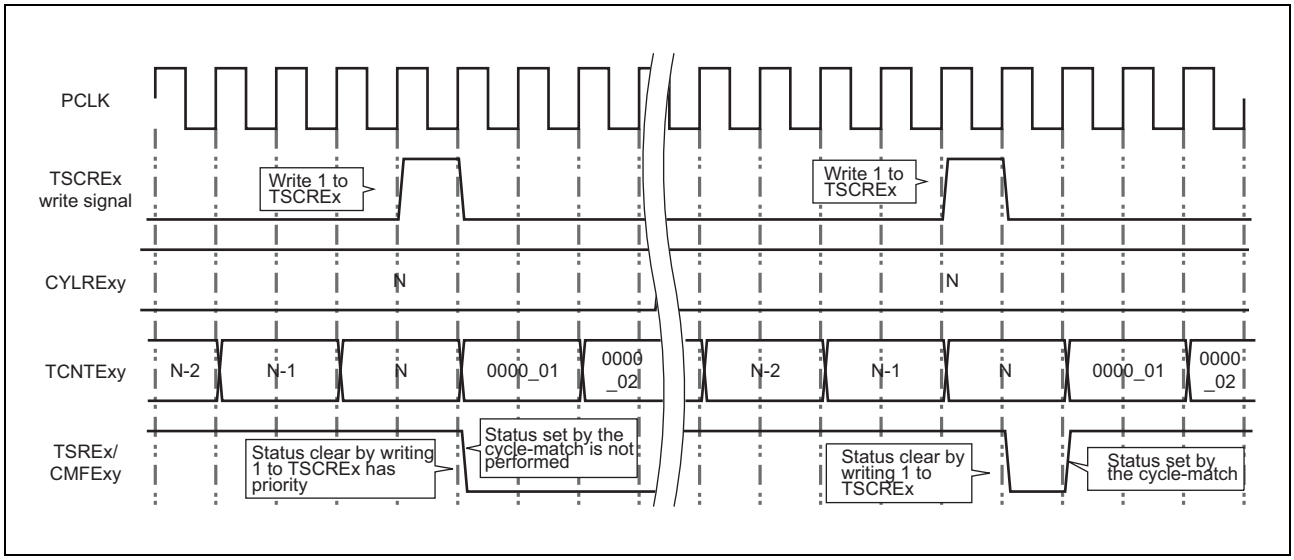


Figure 21.90 Contention between Cycle Match and Cycle Match Status Clear by Writing 1 to the Status Clear Register

Timer	Counter	Compare (Cycle) Match Register	Status
Timer B	TCNTB0	OCRB0	CMFB0
	TCNTB6	OCRB6	CMFB6
	TCNTB6M	ICRB6	CMFB6M
Timer E	TCNTExy	CYLRExy	CMFExy

21.14.2.9 Contention between Detection of 1H Compare Match and Disabling of Counter by ATUENR Setting

When compare match between TCNT1H and OCR1H is detected at the same time a counter enable bit (each bit in ATUENR) is changed to 0, compare match is detected but TCNT2H is not incremented (the left side of **Figure 21.91**). Thereafter, even though a counter enable bit is set to 1, TCNT2H is not incremented until the next compare match occurs.

The waveforms in the right half of **Figure 21.91** show an example in which a counter enable bit is changed to 0 one PCLK cycle after compare match has been detected.

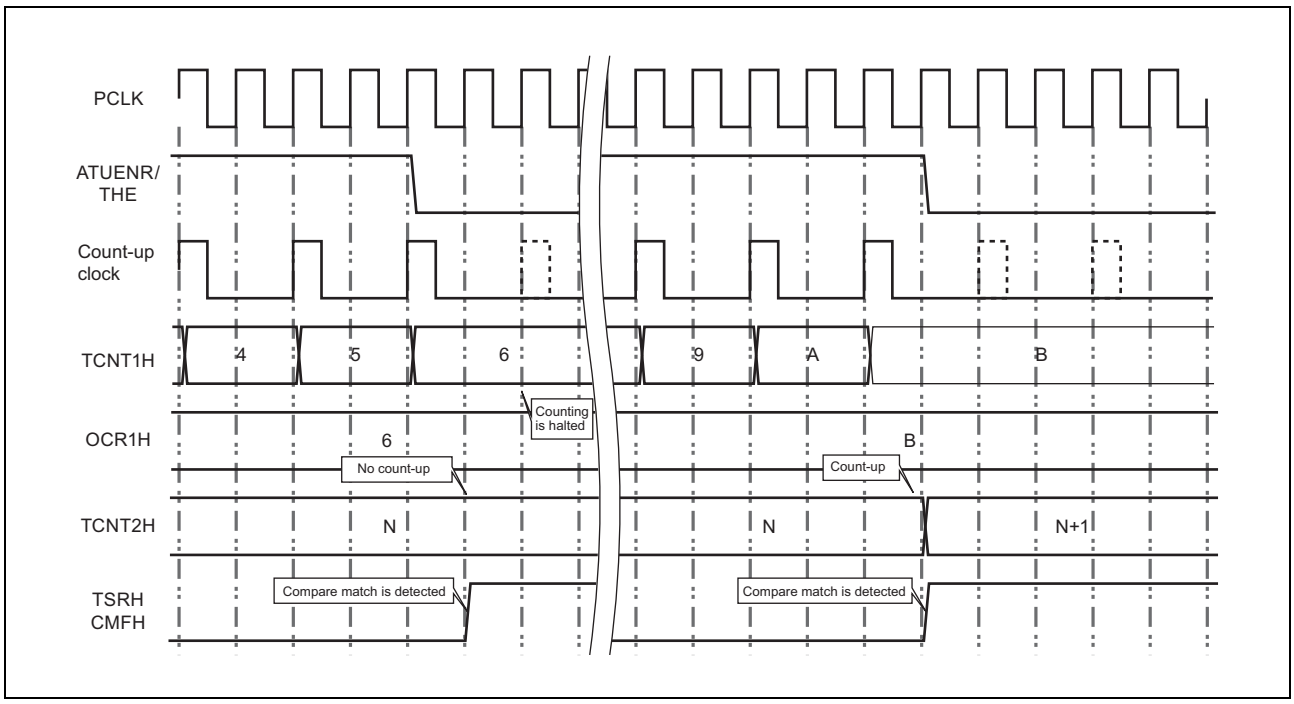


Figure 21.91 Contention between Compare Match Detection and Disabling of Counter

Timer	Counter	Compare Match Register	Status
Timer H	TCNT1H	OCR1H	CMFH

21.14.2.10 Contention between Writing 0 to TCNTExy and Cycle Match

Operation when writing 0 to the 24 higher-order bits of TCNTExy occurs simultaneously with cycle match is shown below. The waveforms in the left half of **Figure 21.92** show a case in which 000 000_H is written to the 24 higher-order bits of TCNTExy at the same time the 24 higher-order bits of TCNTExy is to be cleared to 000 001_H due to cycle match. A cycle match interrupt request is issued and the cycle match status is set, but PWM output does not start because zero writing takes priority. The waveforms in the middle of the figure indicate a case in which 0 is written to the 24 higher-order bits of TCNTExy one PCLK cycle after the counter has been cleared by cycle match. Cycle match detection and PWM output are restarted at the timing the 24 higher-order bits of the TCNTExy counter value changes from N to 1. In contrast with this, the waveforms in the right half of the figure show an example in which 0 is written one PCLK before detection of cycle match. In this case, neither cycle match is detected nor PWM output restarted, and the previous state is retained.

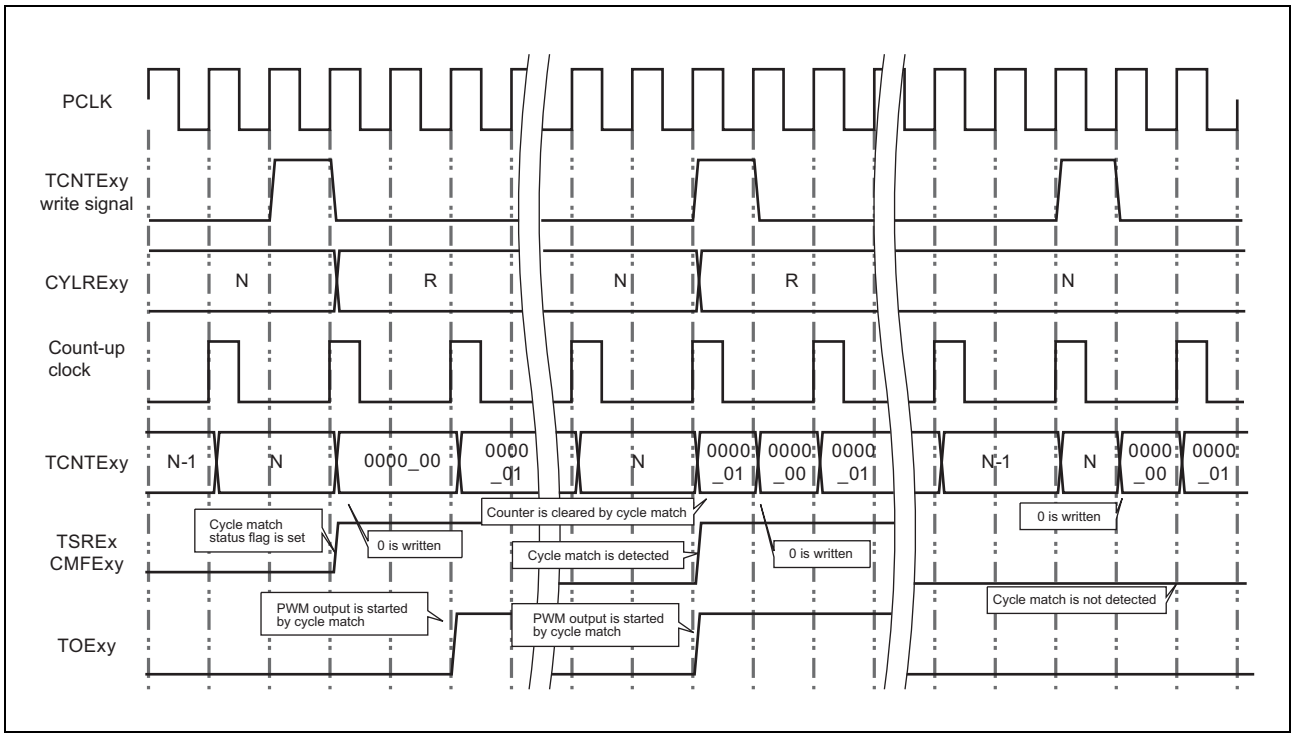


Figure 21.92 Contention between Writing 0 to TCNTExy and Cycle Match

21.14.3 Load/Reload Contention

21.14.3.1 Contention between Data Transfer and Writing to Transfer Destination Register

Contention between data transfer between registers and a peripheral bus write to the transfer destination register is described below. When data transfer occurs simultaneously with writing to the transfer destination register, writing takes priority and the attempt of data transfer is ignored. **Figure 21.93** shows contention between reload to CYLRExy of timer E and writing to it. As shown by the waveforms in the left half of the figure, if writing to CYLRExy occurs at the same timing as cycle reload, writing takes priority. The waveforms in the right half of the figure indicate a case in which CYLRExy is written to immediately after cycle reload.

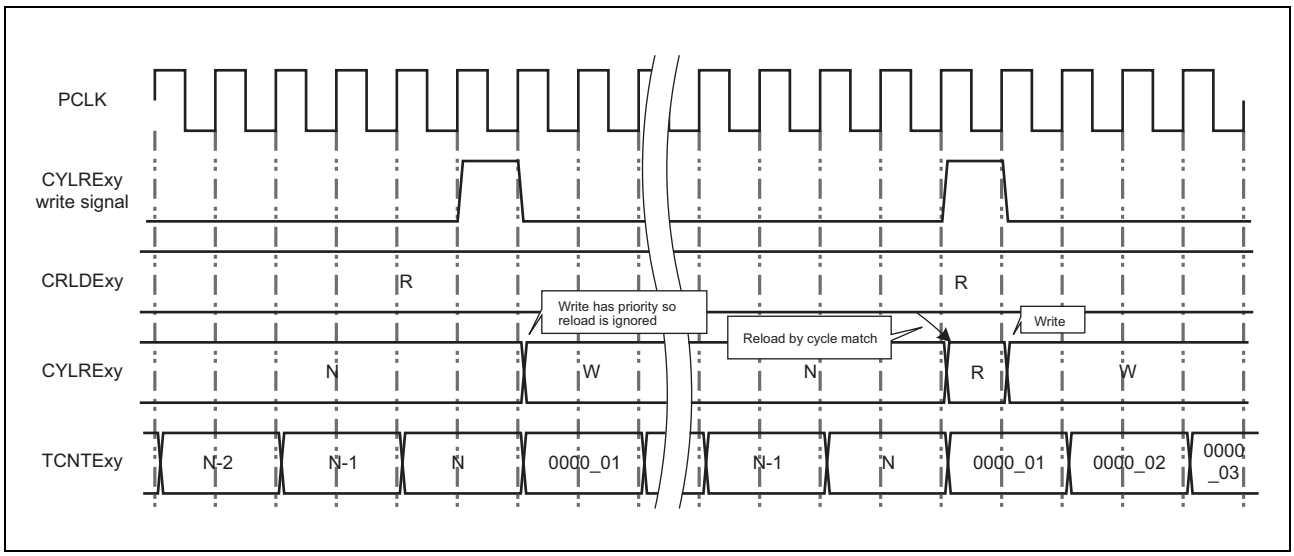


Figure 21.93 Contention between Writing to CYLRExy and Cycle Reload

Timer	Transfer Data	Transfer Destination Register	Transfer Timing
Timer B	ICRB0 LDB TCNTB2 – PIMR TCNTB2 + RLDB	TCNTB2	External event
	ICRB0 – PIMR LDB – PIMR	RLDB	External event
	TCNTB3 + PIMR	TCNTB3	External event
	TCNTB3	TCNTB4	External event
Timer E	CRLDExy	CYLRExy	Cycle match
	DRLDExy	DTRExy	Cycle match

21.14.3.2 Contention between Data Transfer and Writing to Transfer Source Register

Contention between data transfer between registers and a peripheral bus write to the transfer source register is described below. When data transfer occurs simultaneously with writing to the transfer source register, the value prior to writing is transferred. At the same time, the value of the transfer source register is modified. Operation when writing to CRLDExy occurs at the timing of cycle reload is shown below. If writing to CRLDExy occurs at the same timing as cycle reload (waveforms in the left half of **Figure 21.94**), the value immediately before writing is reloaded. On the other hand, the waveforms in the right half of the figure show an example in which CRLDExy is written to one cycle earlier than cycle reload.

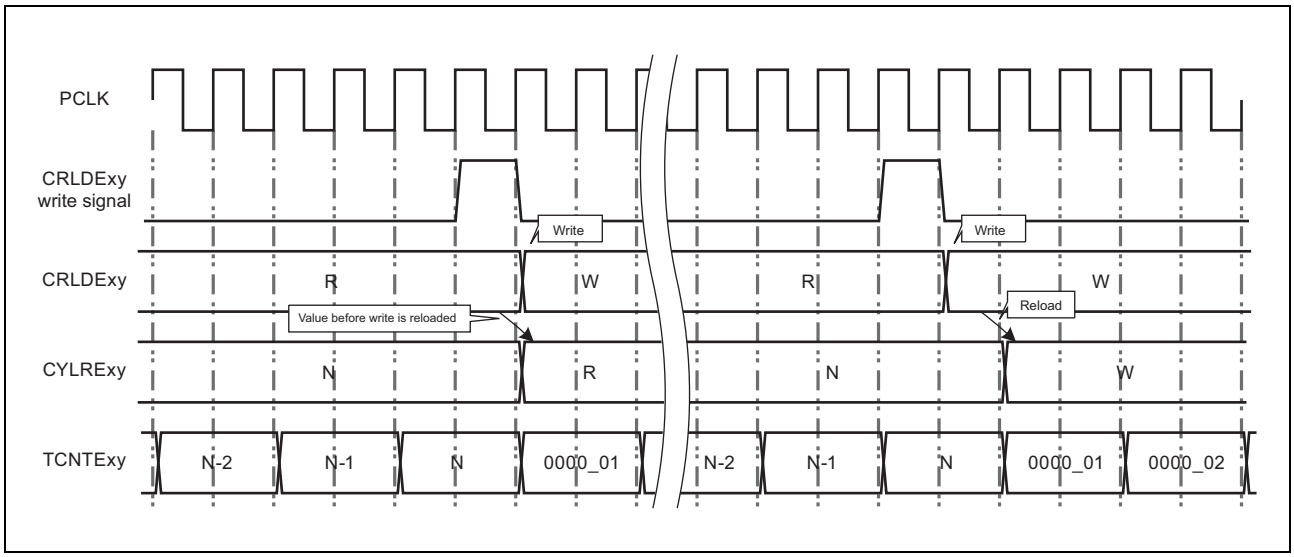


Figure 21.94 Contention between Writing to CRLDExy and Cycle Reload

Timer	Transfer Source Register	Transferred Value	Transfer Destination Register	Transfer Timing
Timer B	LDB	LDB – PIMR	RLDB	External event
		LDB	TCNTB2	External event
	PIMR	ICRB0 – PIMR LDB – PIMR	RLDB	External event
		TCNTB2 – PIMR	TCNTB2	External event
		TCNTB3 + PIMR	TCNTB3	External event
	RLDB	TCNTB2 + RLDB	TCNTB2	External event
	TCNTB3	TCNTB3	TCNTB4	External event
Timer E	CRLDExy	CRLDExy	CYLRExy	Cycle match
	DRLDExy	DRLDExy	DTRExy	Cycle match

21.14.4 Counter Contention

21.14.4.1 Contention between Writing to Counter and Count-Up/Count-Down

When writing to a counter occurs simultaneously with incrementation/decrementation of the counter, the write operation takes priority. The attempt to increment/decrement the value is ignored and incrementation/decrementation recommences from the new value on the next counter clock.

21.14.4.2 Contention between Count-Up and Counter Clearing

When incrementation of a counter occurs simultaneously with clearing of the counter, the counter is not cleared to 0 but cleared to 1.

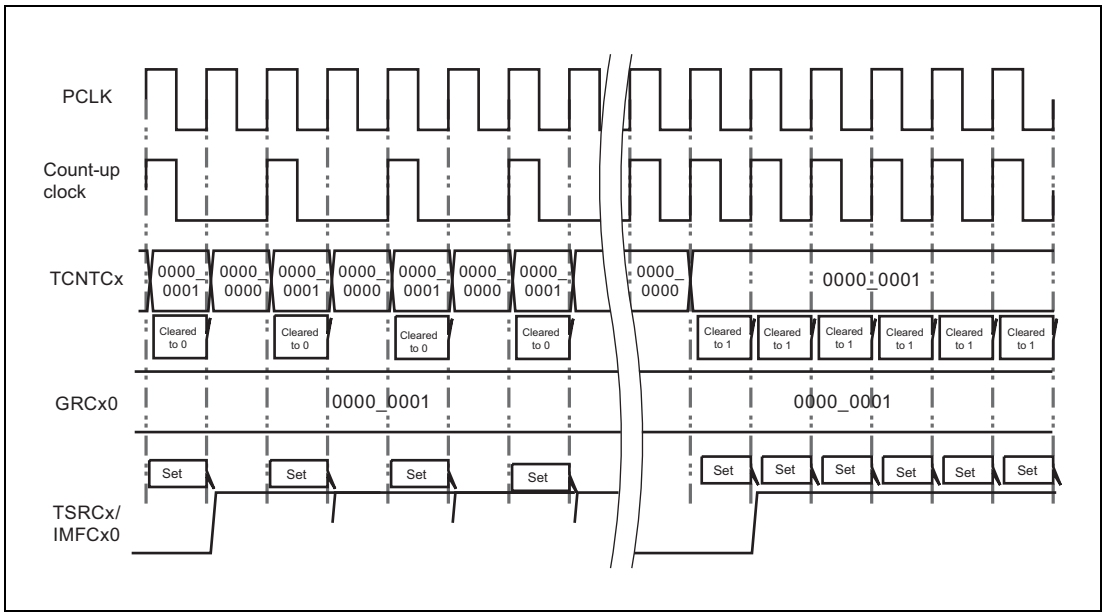


Figure 21.95 Simultaneous Occurrence of Count-Up and Counter Clearing

Timer	Counter	Compare Match Register	Remarks
Timer C	TCNTCx	GRCx0	Only when PWMx0 = 1
	TCNTCx	CUCRCx	Only when PWMx0 = 0, CLRCx = 1 = 1
Timer D	TCNT1Dx	CUCR1Dx	Only when CLR1Dx = 1
	TCNT2Dx	CUCR2Dx	Only when CLR2Dx = 1
Timer F	ECNTAFx	GRAFx	Only when MDFx = 000, 110, or 111
	ECNTBFx	GRBFx	Only when MDFx = 001, 010, or 100
Timer G	TCNTGx	OCRGx	
Timer H	TCNT1H	OCR1H	

21.14.4.3 Contention between Writing to Counter and Overflow

When counter overflow occurs simultaneously with writing to TCNTCx, writing to TCNTCx takes priority. However, an overflow interrupt request is issued, and the overflow status is set (left half of **Figure 21.96**). If the timing for writing to the counter is earlier than incrementation of the counter (waveforms in the right half of the figure), the overflow status flag is not set.

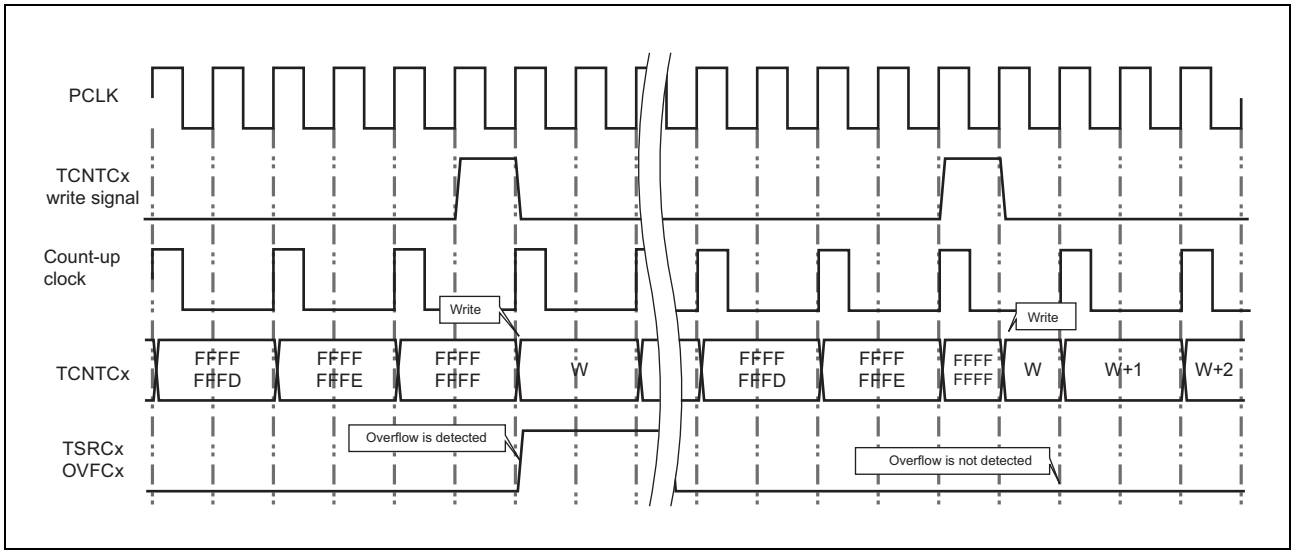


Figure 21.96 Contention between Writing to TCNTCx and Counter Clearing on Overflow

Timer	Counter	Status
Timer A	TCNTA	OVFA
Timer C	TCNTCx	OVFCx
Timer D	TCNT1Dx	OVF1Dx
	TCNT2Dx	OVF2Dx
Timer E	TCNTExy	OVFExy
Timer F	ECNTAFx	OVFAFx
	ECNTBFx	OVFBFx
	ECNTCFx	OVFCFx
Timer G	TCNTGx	OVFGx
Timer H	TCNT1H	OVF1H
	TCNT2H	OVF2H
Timer J	TCNTJx	OVFJx

21.14.4.4 Contention between Setting and Clearing of Overflow Status Flag

When clearing and setting of the overflow status flag occur simultaneously, clearing takes priority. Illustrated below is an example where status flag setting by an overflow of the counter value (FFFF $FFFF_H$ to 0000 0000_H) and flag clearing by the timer status clear register occur simultaneously (left half of **Figure 21.97**). The waveforms in the right half of the figure show the way the overflow status flag is set again immediately after the status flag has been cleared.

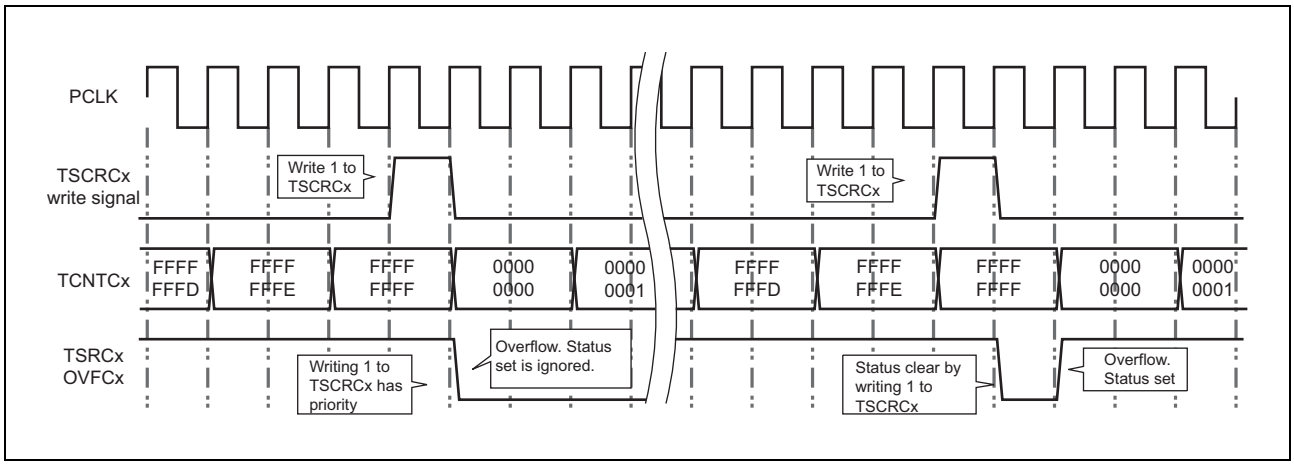


Figure 21.97 Contention between Setting and Clearing of Overflow Status Flag

Timer	Counter	Status
Timer A	TCNTA	OVFA
Timer C	TCNTCx	OVFCx
Timer D	TCNT1Dx	OVF1Dx
	TCNT2Dx	OVF2Dx
Timer E	TCNTExy	OVFExy
Timer F	ECNTAFx	OVFAFx
	ECNTBFx	OVFBFx
	ECNTCFx	OVFCFx
Timer G	TCNTGx	OVFGx
Timer H	TCNT1H	OVF1H
	TCNT2H	OVF2H
Timer J	TCNTJ x	OVFJx

21.14.4.5 Contention between Overflow and Counter Clearing by Compare Match

If the maximum value is set in a compare match register that has the function to clear a counter by compare match and that function is enabled, when the counter reaches its maximum value, the counter is cleared. At this time, an overflow interrupt request is not issued even if the counter clock is 1/1PCLK, so that an overflow is not detected.

Examples using TCNT1H and OCR1H of timer H are shown in **Figure 21.98**. With $FFFF_H$ set in OCR1H, a case in which the count-up clock is not PCLK × 1/1 (waveforms in the left half of the figure) and a case in which the count-up clock is PCLK × 1/1 (waveforms in the right half of the figure) are shown.

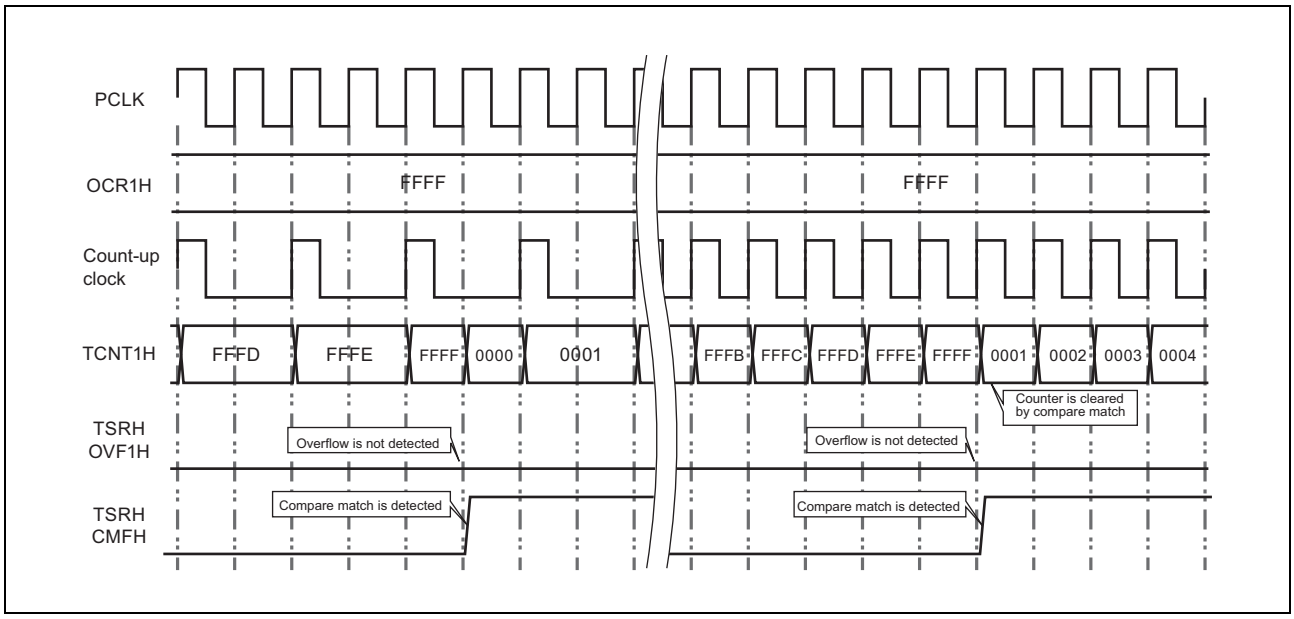


Figure 21.98 Contention between TCNT1H Counter Overflow and Compare Match

Timer	Counter	Status	Remarks
Timer C	TCNTCx	OVFCx	Only when PWMn0 = 1
Timer F	ECNTAFx	OVFAFx	Only when MDFx = 000, 110, or 111
	ECNTBFx	OVFBFx	Only when MDFx = 001
Timer G	TCNTGx	OVFGx	
Timer H	TCNT1H	OVF1H	

If the compare match counter clear feature is not provided or is disabled, an overflow interrupt request is issued and the overflow status is set. **Figure 21.99** shows operation when the PWMx0 bit of timer C is 1 (counter clearing is enabled) and when the PWMx0 bit is 0 (counter clearing is disabled).

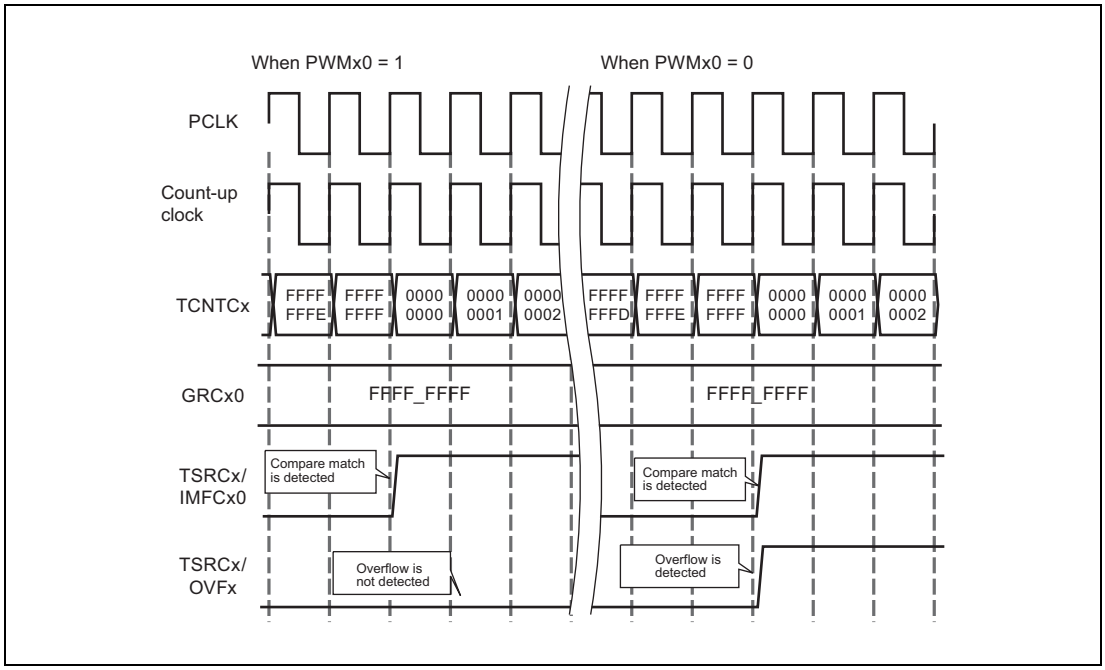


Figure 21.99 Contention between Counter Clearing by Compare Match of Timer C and Overflow (PWMx0 = 1/0)

21.14.5 Contention in Noise Canceler

Contentions in the noise cancelers are described here.

Timer	Counter	Compare Match Register
Timer A	NCNTAx	NCRAx
Timer C	NCNTCxy	NCRCxy
Timer F	NCNTAFx	NCRAFx
	NCNTBFx	NCRBFx
Timer J	NCNTJx	NCRJx

21.14.5.1 Contention between Writing to Noise Canceler Counter and Compare Match with Noise Canceler Register

When writing to NCNT occurs simultaneously with a compare match with NCR, writing takes priority. An example in minimum time-at-level cancellation mode is shown below. In the example in the left half of **Figure 21.100**, since writing prevents compare match from occurring, input capture is also not performed. The example in the right half of the figure shows a case in which writing is performed one PCLK cycle later. In this case, compare match occurs so input capture processing is carried out.

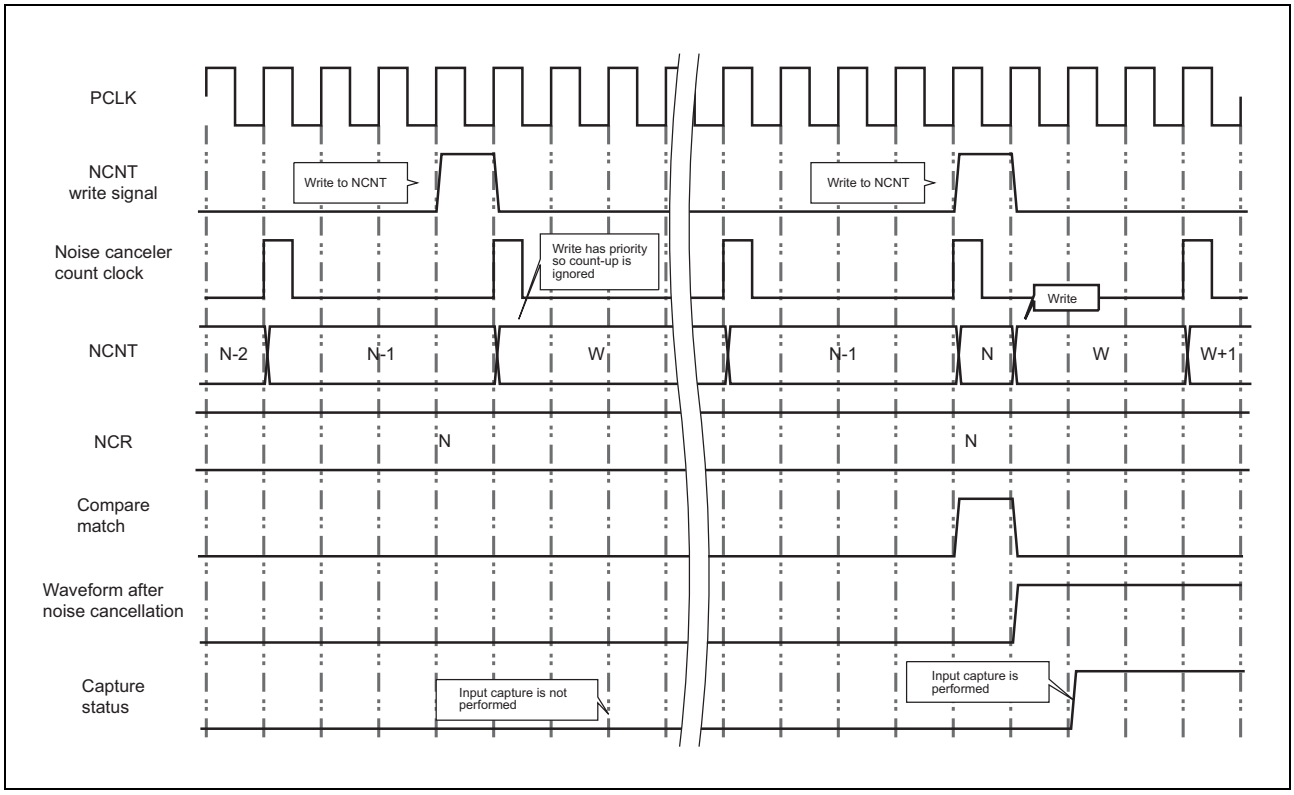


Figure 21.100 Contention between Writing to NCNT and Compare Match of NCN-TNCR (Example in Minimum Time-at-Level Cancellation Mode)

21.14.5.2 Contention between Writing to Noise Canceler Register and Compare Match with Noise Canceler Counter

When writing to NCR occurs simultaneously with a compare match with NCNT, writing takes priority. An example in minimum time-at-level cancellation mode is shown below. In the example in the left half of **Figure 21.101**, since writing prevents compare match from occurring, input capture is also not performed. The example in the right half of the figure shows a case in which writing is performed one PCLK cycle later. In this case, compare match occurs so input capture processing is carried out.

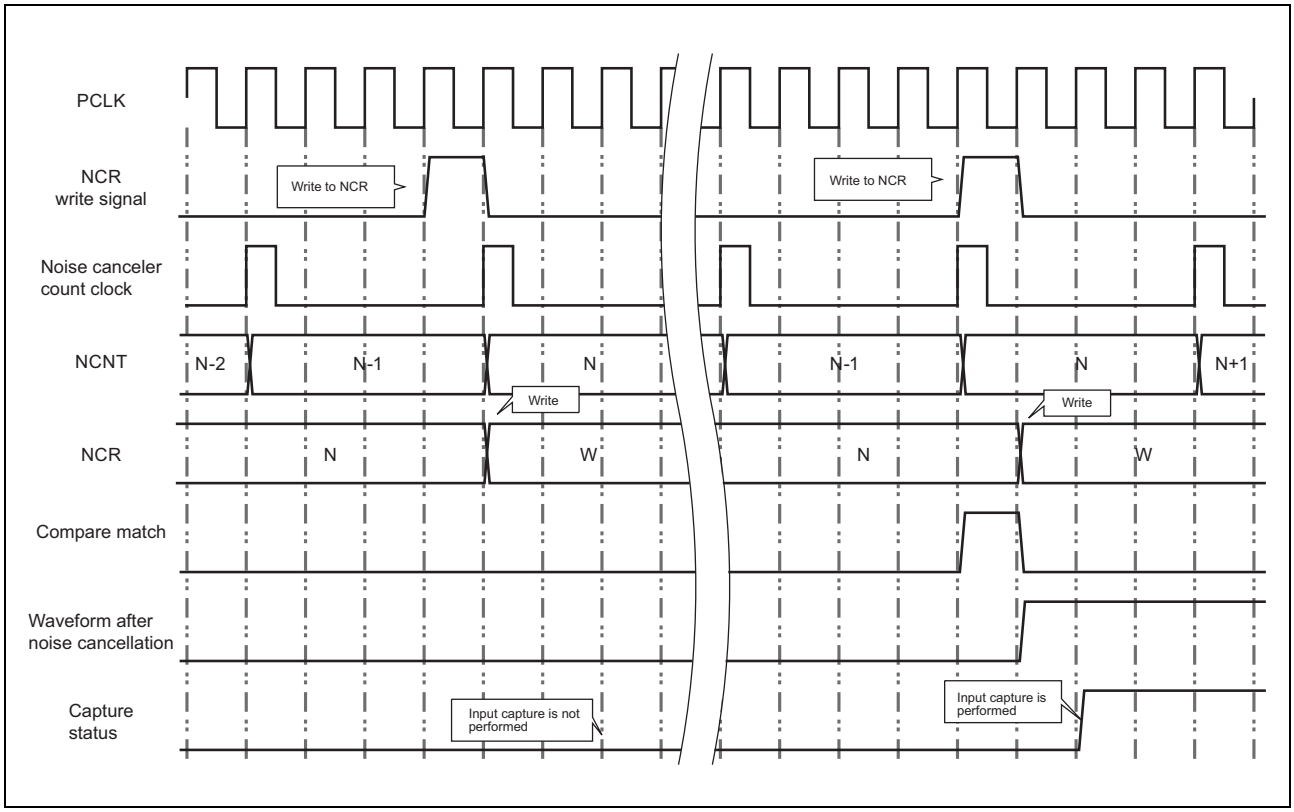


Figure 21.101 Contention between Writing to NCR and Compare Match of NCR-NCNT (Example in Minimum Time-at-Level Cancellation Mode)

21.14.6 Contention in Timer Down Counter Dxy

Contentions in DCNTDxy are described here.

21.14.6.1 Contention between Writing to DCNTDxy Counter and Count-Down

When writing occurs simultaneously with decrementation of the down counter, writing to DCNTDxy is performed. The attempt to decrement the value is ignored and decrementation recommences from the new value on the next count-down clock.

21.14.6.2 Contention between Writing to DCNTDxy Counter and Underflow

When writing to DCNTDxy occurs simultaneously with underflow, writing to DCNTDxy is performed. The example in the left half of **Figure 21.102** shows operation when a count-down clock is input simultaneously with writing to DCNTDxy when the DCNTDxy value is 0000 0000_H. Though the new value is written to DCNTDxy, count-down operation will be halted because underflow is detected. The underflow status flag is set. Figure on the right illustrates the case where data is written to DCNTDxy one cycle before, in which case an underflow interrupt request is not issued, so that an underflow will not be detected.

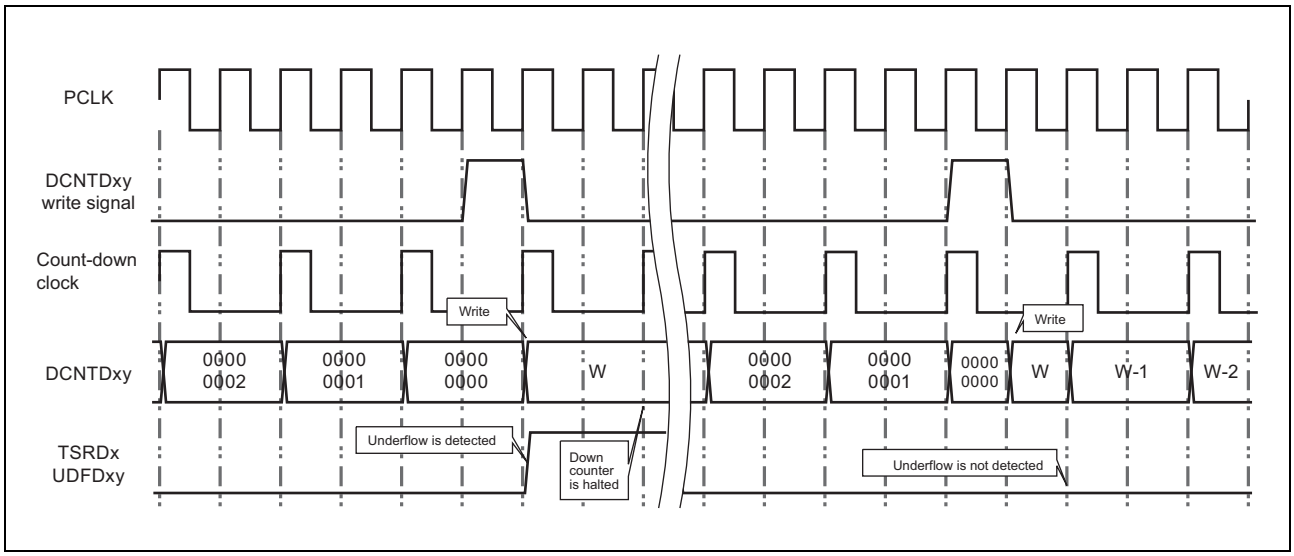


Figure 21.102 Contention between Writing to DCNTDxy and Underflow

21.14.6.3 Counter Stop Trigger — Contention between Writing to DCNTDxy Counter and Compare Match B

When writing to DCNTDxy occurs simultaneously with compare match B, writing to DCNTDxy is performed (if compare match B is selected as a condition to stop the down counter).

An example in which writing to DCNTDxy occurs simultaneously with clearing of the counter by detection of compare match B is shown in the middle of **Figure 21.103**. Counter clearing of DCNTDxy by compare match B is ignored, and writing takes priority. However, the output on TODxyB is turned off due to compare match B, and DCNTDxy halts with the written value retained. The waveforms in the right half of the figure show a case in which the write cycle occurs one PCLK cycle earlier. During the PCLK cycle subsequent to writing to DCNTDxy, the counter is cleared by compare match B. The waveforms in the left half of the figure show an example in which writing is performed immediately after the counter has been cleared by compare match B.

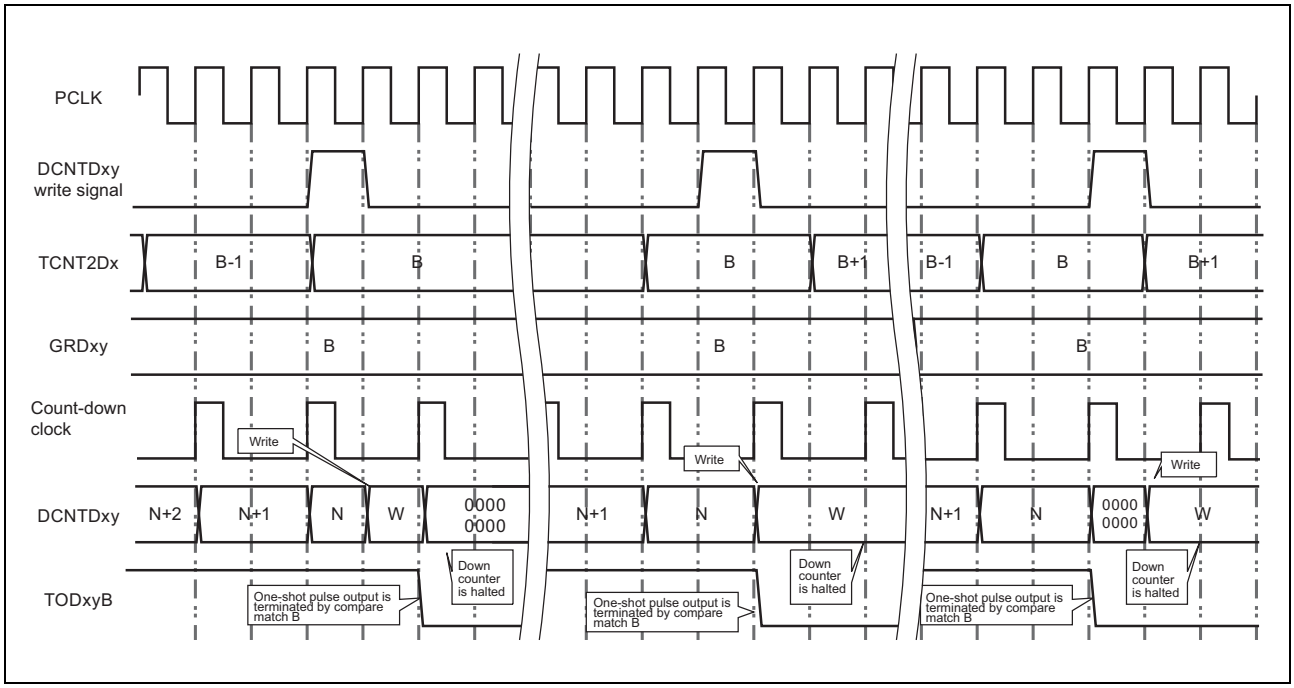


Figure 21.103 Contention between Writing to DCNTDxy and Counter Clearing by Compare Match B

21.14.6.4 Contention between Setting of Underflow Status Flag and Clearing by Writing 1 to Status Clear Register

When flag clearing by the timer status clear register and an underflow occurs simultaneously, flag clearing is performed. Illustrated below is an example where status flag setting by an underflow of DCNTDxy and flag clearing by the timer status clear register occur simultaneously (left half the figure below). The waveforms in the right half of **Figure 21.104** show how the status flag is set again by underflow occurrence immediately after the status flag has been cleared.

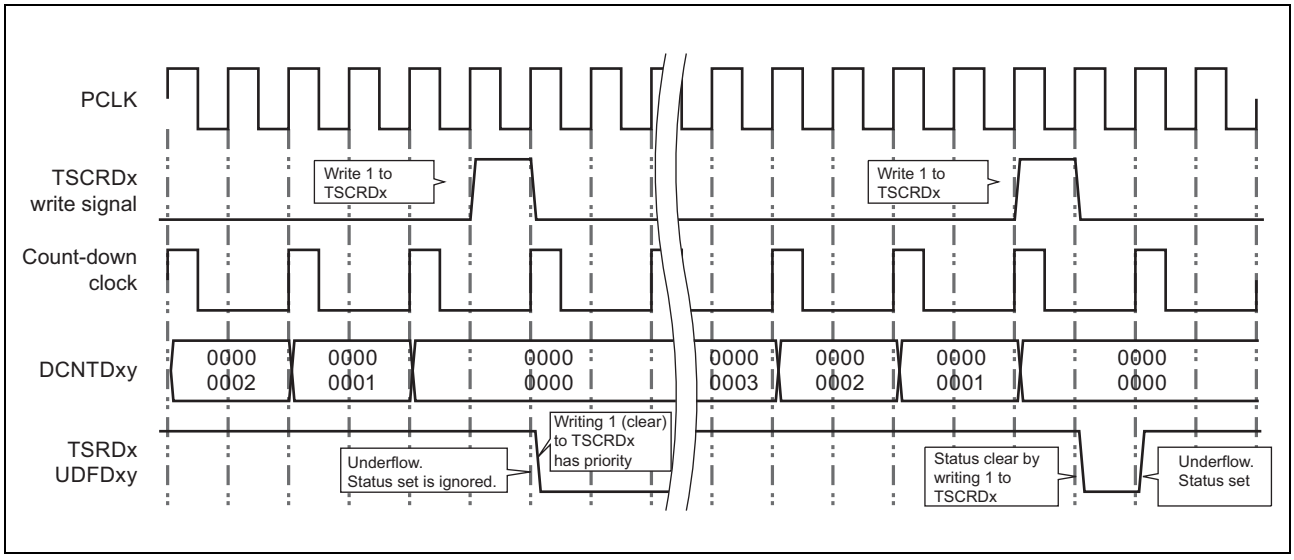


Figure 21.104 Contention between Setting and Clearing of Underflow Status Flag

21.14.6.5 TODxyB Output at Occurrence of Down Counter Start Trigger When Down Counter Value is 0000 0000$_H$

The TODxyB output (one-shot pulse) is not started by down counter underflow.

21.14.6.6 TODxyB Output at Simultaneous Occurrence of Start Trigger and Stop Trigger for Down Counter

When the down counter start trigger occurs simultaneously with the down counter stop trigger, the down counter is cleared to 0 by the stop trigger. In this case, the one-shot pulse output on the TODxyB pin is not started.

Similar to the above case, when the down counter start trigger occurs and then the down counter stop trigger occurs before the first count-down clock is input, the down counter is cleared to 0 without being decremented even once and the one-shot pulse output on the TODxyB pin is not started.

21.14.6.7 Contention between Down Counter Start Trigger and Underflow

When the down counter start trigger occurs simultaneously with underflow, DCNTDxy remains halted at the value of 0000 0000_H (waveforms in the middle of **Figure 21.105**). If the down counter had been in process of decrementing, the TODxyB output is turned off by underflow (waveforms in the left half of the figure). If compare match A occurs while the down counter is halted (DCNTDxy = 0000 0000_H), the TODxyB output is kept negated (waveforms in the right half of the figure). In both cases, the underflow interrupt request and setting of the underflow flag to 1 are performed upon detection of a down count start trigger or at the first cycle of the down count clock after detection.

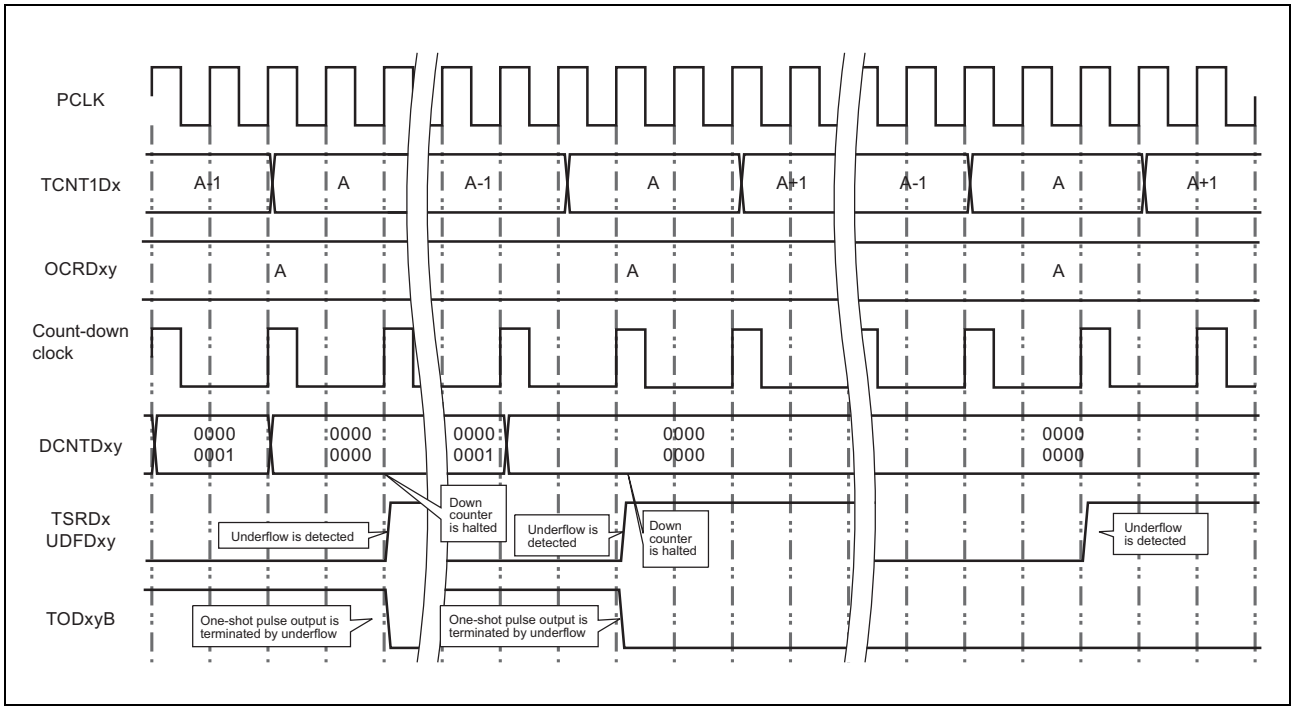


Figure 21.105 Contention between Compare Match A and Underflow

21.14.7 Coordinated Operation of Timers A, B, and D

This section explains the contention between counter clear requests from timer B and TCNT1Dx and TCNT2Dx.

21.14.7.1 Contention Between Counter Clearing of TCNT1Dx and 2Dx and Compare Match

Operation when clearing of the TCNT1Dx/TCNT2Dx counter by a counter clearing signal from timer B occurs simultaneously with compare match is shown below. The waveforms in the left half of **Figure 21.106** show a case in which the counter is cleared prior to compare match. On the other hand, the waveforms in the right half of the figure show a case in which the counter is cleared simultaneously with compare match.

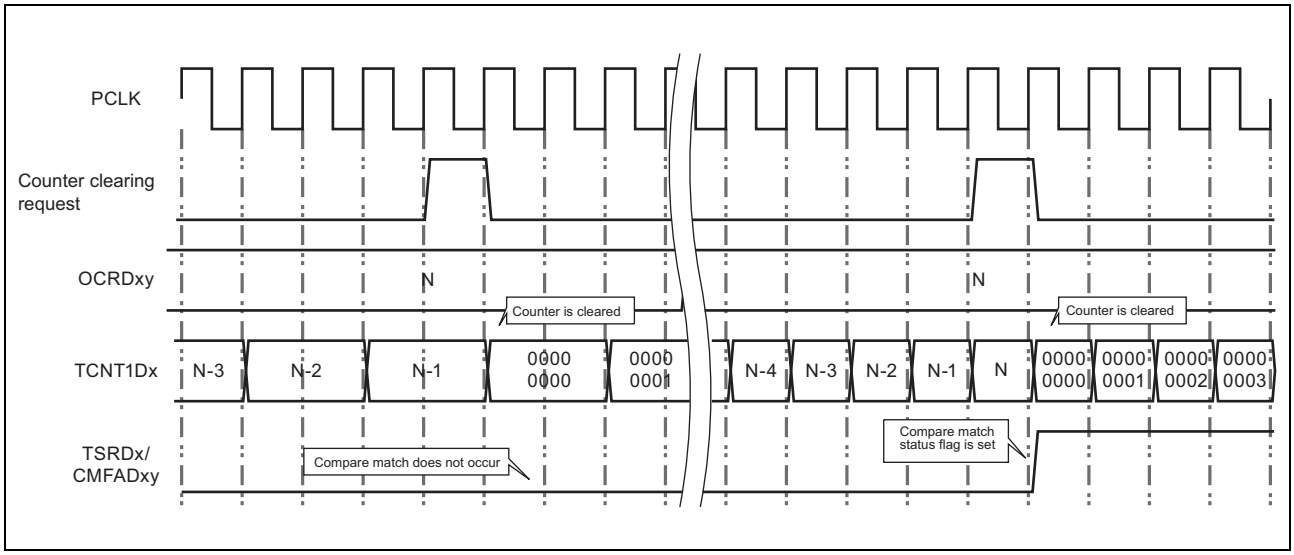


Figure 21.106 Contention between Counter Clearing by Timer B and Compare Match

Timer	Counter	Counter Clearing Source	Compare Match Register	Status
Timer D	TCNT1Dx	TCNT1Dx/TCNT2Dx clearing request from timer B	OCR1Dxy	CMFADxy
	TCNT2Dx		OCR2Dxy	CMFBDxy

21.14.7.2 Contention between TCNT1Dx/TCNT2Dx Counter Overflow and Counter Clearing by Timer B

When writing to TCNT1Dx/TCNT2Dx occurs simultaneously with a counter clearing signal from timer B, the counter is not cleared but writing to the counter is performed (waveforms in the left half of **Figure 21.107**). The waveforms in the right half of the figure show a case in which writing to TCNT1Dx is one PCLK cycle later. This is the same for TCNT2Dx.

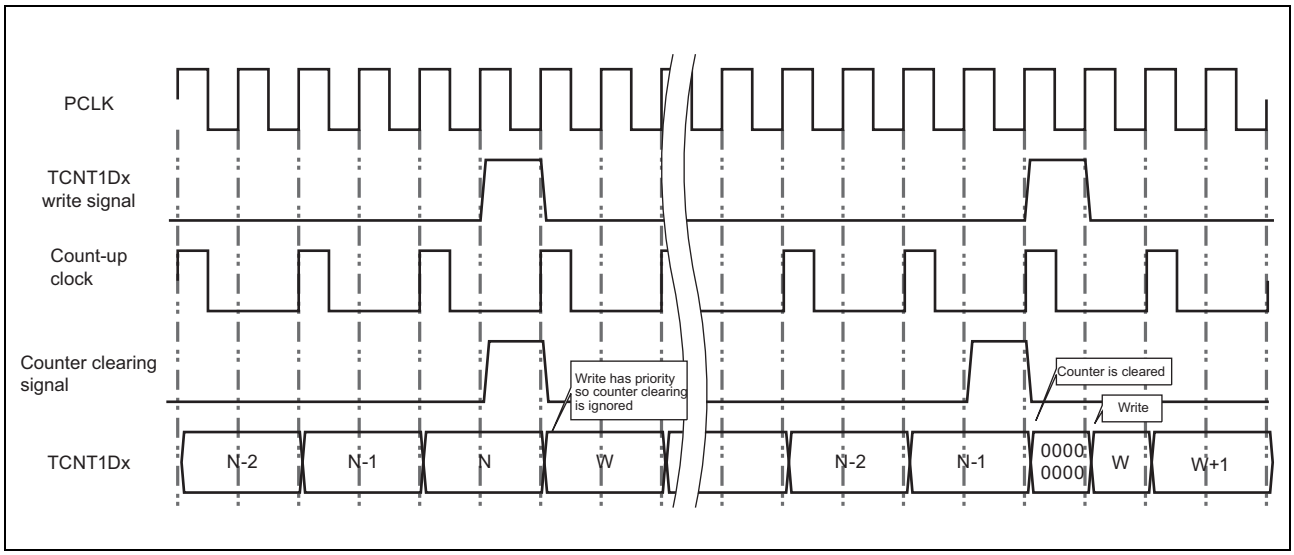


Figure 21.107 Contention between Writing to TCNT1Dx and Counter Clearing

21.14.7.3 Contention between TCNT1Dx/TCNT2Dx Counter Overflow and Counter Clearing by Timer B

When TCNT1Dx overflow by the counting up occurs simultaneously with clearing of TCNT1Dx from timer B, the counter value is cleared to 0000 0000_H by the counter clearing signal. In this case, the overflow status flag is not set (only for when C1CEDx = 1). The same applies to TCNT2Dx overflow.

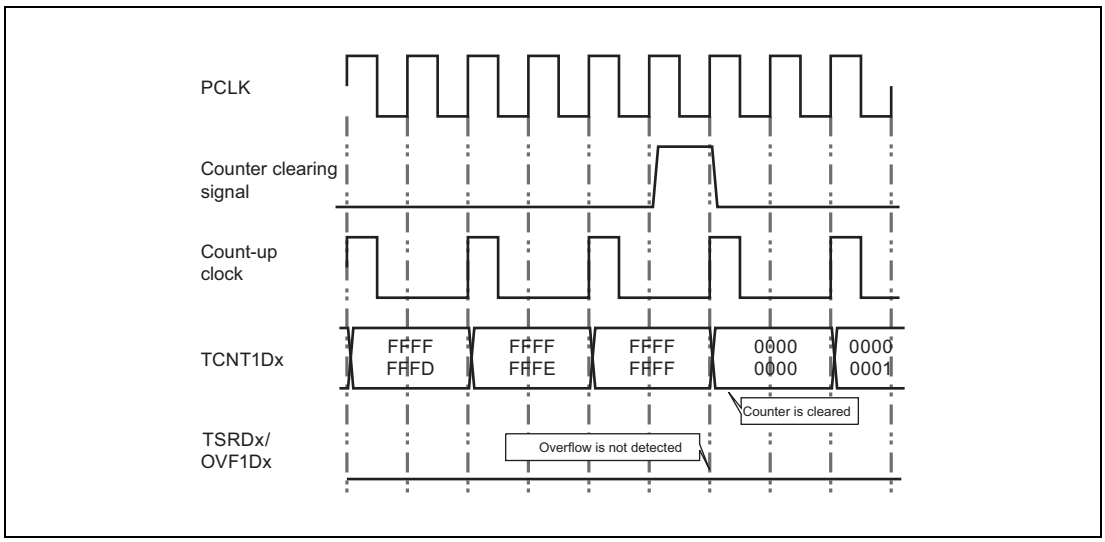


Figure 21.108 Contention between Counter Clearing and Overflow

21.14.7.4 Contention between TCNT1Dx Clearing by Clearing Signal from Timer B and Input Capture to OSBRDx

Operation when clearing of the TCNT1Dx counter by timer B occurs simultaneously with input capture to an offset base register is shown below. When capture and counter clearing occur simultaneously, the counter value before clearing is captured in OSBRDx. At the same time, the TCNT1Dx counter is cleared (waveforms in the left half of **Figure 21.109**).

The waveforms in the right half of the figure show a case in which capture is performed one PCLK cycle after the counter has been cleared and the counter value after clearing is captured in OSBRDx.

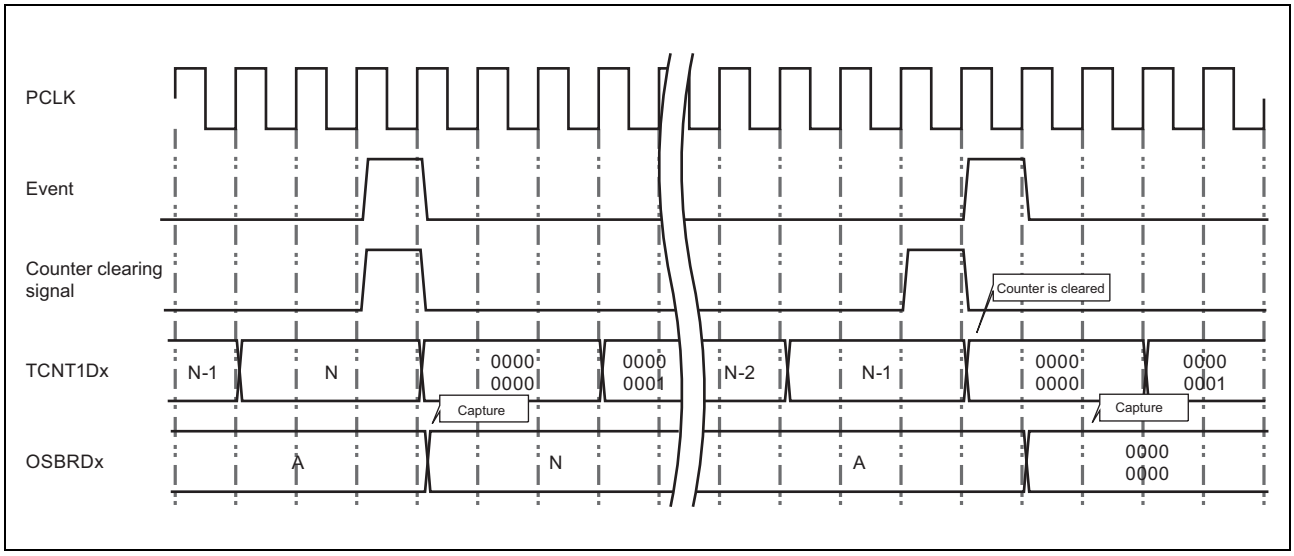


Figure 21.109 Contention between TCNT1Dx Counter Clearing and Input Capture to OSBRDx

21.14.7.5 Contention between Clearing of the Counter to 0 by Timer B and Clearing of the Counter by the Counter Upper-Limit Setting Function

If the signal for clearing the counter to 0 from timer B and clearing by the timer D counter upper-limit setting function are in contention, the former is given priority and is responsible for the counter value becoming 0.

Applicable Timer	Counter	Counter Clearing Source	Compare-Match Register
Timer D	TCNT1Dx	Request for clearing TCNT1Dx from timer B	CUCR1Dx
	TCNT2Dx	Request for clearing TCNT2Dx from timer B	CUCR2Dx

21.14.8 Different Specifications of Operation in Response to a Match in Comparison

Operation in response to a match in comparison differs from timer to timer. Timers have the following three types of operation, according to differences in the time the match in comparison is detected and the condition for detection.

- Type 1
  - The match in comparison is detected in the cycle of PCLK following that in which the values of the timer counter and the compare-match register have matched.
  - Testing for a match in comparison proceeds in every cycle of PCLK.
  - A match in comparison is detected even if it occurs at the time a value is written to the timer counter or the compare-match register.
- Type 2
  - The match in comparison is detected in the cycle of PCLK following that in which the values of the timer counter and the compare-match register have matched.
  - A match in comparison is detected in the cycle of PCLK when the timer counter counts up or down to the matching value.
  - A match in comparison is not detected if it occurs at the time a value is written to the timer counter or the compare-match register.
- Type 3
  - A match in comparison is detected in the next cycle of the counter clock after the values of the timer counter and the compare-match register have matched.
  - Testing for a match in comparison proceeds in every cycle of the counter clock.
  - A match in comparison is not detected if it occurs at the time a value is written to the timer counter or the compare-match register.

Applicable Timer	Counter	Compare-Match Register	Type of Match in Comparison
Timer B	TCNTB0	OCRB0	Type 3
	TCNTB1	OCRB1	Type 1
		OCRB10	Type 1
		OCRB11	Type 1
		OCRB12	Type 1
	TCNTB6	OCRB6	Type 3
		OCRB7	Type 3
	TCNTB3	OCRB8	Type 1
Timer C	TCNTCx	CUCRCx	Type 1
		GRCxy	Type 1
		OCRCxy	Type 1
Timer D	TCNT1Dx	CUCR1Dx	Type 1
		OCR1Dxy	Type 2
	TCNT2Dx	CUCR2Dx	Type 1
		OCR2Dxy	Type 2
Timer E	TCNTExy	CYLRExy	Type 3
		DTRExy	Type 3

Applicable Timer	Counter	Compare-Match Register	Type of Match in Comparison
Timer F	ECNTAFx	GRAFx	Type 1
	ECNTBFx	GRBFx	Type 1
	ECNTCFx	GRCFx	Type 1
		GRDFx	Type 1
Timer G	TCNTGx	OCRGx	Type 1
Timer H	TCNT1H	OCR1H	Type 1
Timer J	TCNTJx	OCRJx	Type 1

Section 22 Autonomous Pulse Adapter (APA)

The autonomous pulse adapter (APA) enables real-time generation of complex pulse waveforms without intervention by the CPU. Use of APA reduces the workload on the CPU and periphery buses, and coordinates the PWM channels and the reference inputs for pulse generation.

22.1 Overview

Basic functions

- Real-time generation of complex waveforms without intervention by the CPU.
- Built-in calculation device for real-time calculation of pulse width.
- Generation of complex pulse sequences by defining one-shot pulses, which are basic building blocks.
- A one-shot pulse is generated from multiple instances of reference data. Times, angles, and AD conversion values can be combined.
- Pulse sequences can be controlled (can be put to different states) on a real-time basis by event inputs. The manner of control by event inputs can be selected from multiple options (stop of output, transition to the next pulse, output masks, etc.)
- Synchronization of multiple channels by using the same event signal to control the channels.

Reference input

- The following 3 types (11 channels) can be used as reference inputs, each of which has a different time resolution.

CAUTION

SAR-AD cannot be used as APA reference in this product.

Table 22.1 Reference Input

Type	Time Resolution	Number of Channels	Dynamic Range	Remarks
Reference input A	pclk *4	2 (ch.0 and ch.1)	16 bits	External input
Reference input C	pclk *20	8 (ch.2 to ch.9)	16 bits	External input
Reference input S	pclk *20	1 (ch.sw)	16 bits	These registers can be used as reference inputs, and can be read and written by software.

Note: A signal to be used as a reference input must have slower update intervals than time resolutions in **Table 22.1** (If it is faster, APA cannot capture data inside, resulting in data losses.)

Table 22.2 Example of Application (electric current feedback configuration pclk = 40 MHz)

Type	Time Resolution	Channels	Dynamic Range	Reference Used	Connected to	Remarks
Timer	10 MHz (pclk*4)	ch.0, ch.1	16 bits	Reference A *2	ATU (Timer D, Timer G)	ATU output is 32 bits, and 16 bits of them are used. It is used by the register of the APA input selector (see **Section 22.5.8, APA Input Selector**).
ADC	2 MHz (pclk*20)	ch.2 to ch.9	16 bits	Reference C *8	ΔΣ-ADC or SAR-AD	—
Software reference	2 MHz (pclk*20)	ch.sw	16 bits	Reference S *1	—	—

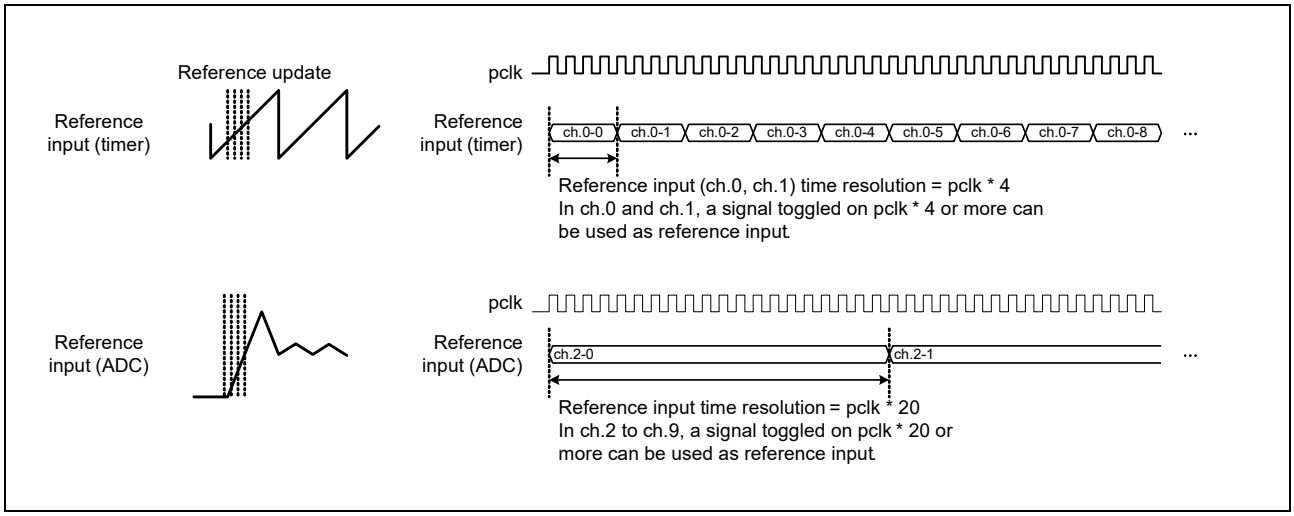


Figure 22.1 Use Conditions of Reference Inputs

Event input

- 32 channels are available
  - Selection of 16 channels is possible from 64 channels of event inputs (including 6 channels of software events).
  - For other 16 channels, PWM output generated by APA can be used as event input.
- Event input to pulse generation channels is performed by time division.

Table 22.3 Event Input

Type	Time Resolution	Number of Channels	Detection Edge	Remarks
Event input	pclk * 1	58	Rise, fall, or rise and fall	—
Software event input	pclk * 1	6	Rise, fall, or rise and fall	These registers can be used as event inputs, and can be read and written by software.

Pulse generation channels

- PWM pulse generation can use 16 channels.
- Each pulse generation channel can have 1 interrupt output channel (16 channels in total).
  - An interrupt cause can be one of the following 5 types: element transition, channel operation completion, on match, off match, and on/off match.
- Pulse generation is realized by setting a set of parameters (called an element). Each parameter specifies a PWM output enabling condition (=, <, or >) for a reference input, a PWM output disabling condition (=, <, or >) for a reference input, or exception processing for an event input.
- Combination of multiple elements can realize a complex pulse sequence.

Table 22.4 Pulse/Interrupt Output

Type	Time Resolution	Number of Channels	Minimum Pulse Width	Remarks
PWM output*1	pclk * 1	16	pclk * 40 (High) pclk * 40 (Low)	The minimum pulse width shown here is applicable when ON/OFF occurs on the match condition specified by an element. If the match status is forcibly changed by exception processing, a shorter pulse width may result from the conflict with a match.
Interrupt output	pclk * 1	16	pclk * 1 (High)	Only one-shot pulse (high) can be output.

Note 1. PWM output matching condition can be specified by an element. A matching condition must satisfy the minimum pulse widths listed in the above table (see **Section 22.5.7.1, Matching Condition Setting 1**). Specifying a matching condition with a pulse width of pclk*40 or shorter might result in an unintended pulse width.

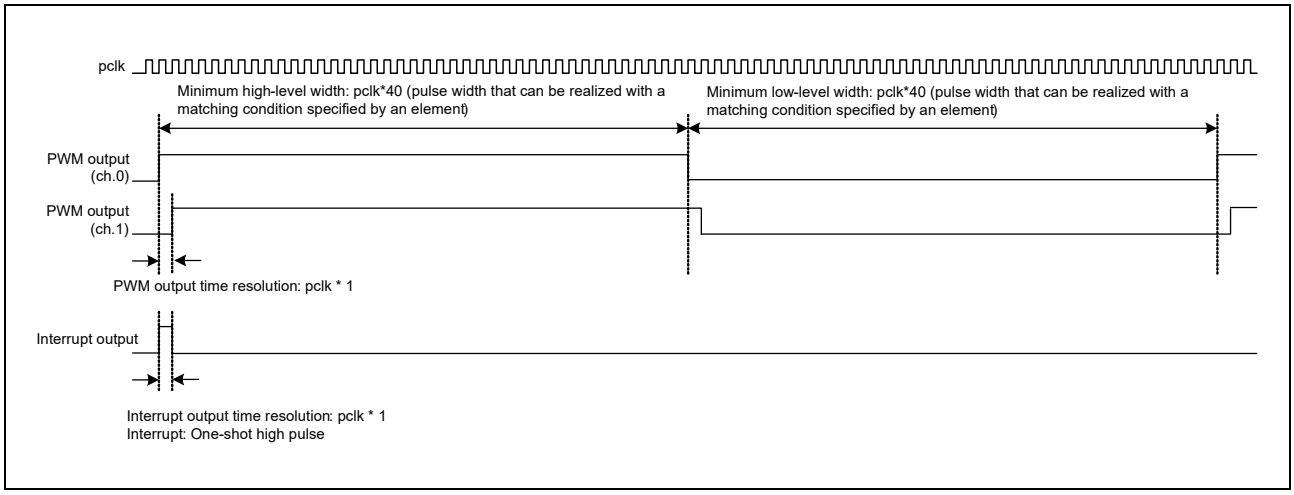


Figure 22.2 Use Condition of Pulse/Interrupt Output

Element RAM Interface

- In APA, 128 elements can be defined.
- A piece of element information can be shared by all pulse generation channels (16 channels). Access from a pulse generation channel to an element RAM is made with time division which is controlled by the event input block.
- Element data is written before a pulse generation channel generates pulses. (through the P-Bus).
- Element data can be read through P-Bus.
- Each element is stored in memory, and can be read and written through the P-Bus. Writing to an element through the P-Bus must be made successively in units of 3 words (32 bits * 3).
- Built-in capability of calculating matching conditions (relative/absolute).

22.2 Terms and Definitions

Table 22.5 lists and defines the terms used in this section.

Table 22.5 List of Terms (1/2)

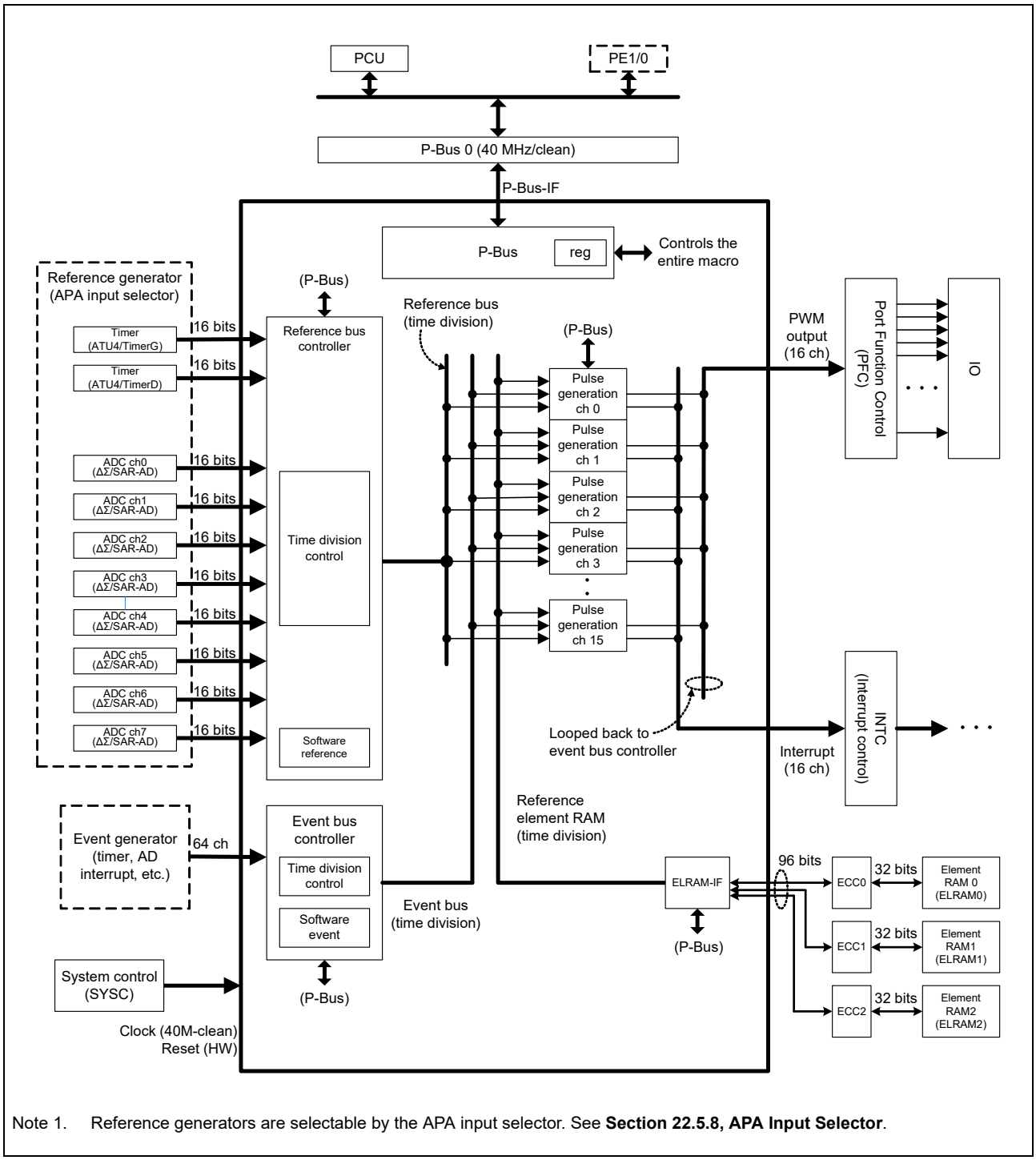
Terms	Definition in This Section
APA	Abbreviation of Autonomous Pulse Adapter. Autonomous pulse control adapter.
External reference	Various data output signals to be referenced for pulse generation such as timers, ADCs, and sensors are referred to as “references” here. The configuration here is intended to allow 16-bit, 10-channel references for each channel. References of different resolutions (frequencies) can coexist.
Software reference	A register area (16 bits) that can be used as a virtual reference through writing data by software. One channel is available.
Reference bus	An APA-internal local bus that multiplexes all 11 reference channels (10 external and 1 software).
Event	A signal that triggers an APA-internal control action is referred to as an “event” here. Examples of events include an interrupt signal from peripheral modules and PWM output signal from APA itself. Events are represented by 1 bit per channel. Up to 32 channels can be used (among which 16 channels can be selected from 64 input channels, and other 16 channels have fixed connection).
Software event	A register area that can be used as a virtual event through writing data by software. 1 bit * 6 channels are available.
Event bus	An APA-internal local bus that connects 32 event channels in a parallel manner (eb_dat_r[31:0]).
Element	A data structure of a set of parameters as conditions for pulse generation. Each pulse generation channel generates pulses in accordance with the setting defined by the referenced element. • An element contains such information as a reference input number (0 to 7), a PWM output enabling condition (=, <, or >), a PWM output disabling condition (=, <, or >), exception processing for an event input, and a pointer to the next element. • An element is deployed in an APA-internal local memory (element RAM), and is configured through the P-Bus before pulse generation. • Each element has a pointer to the next elements, which is used in exception processing (transfer) to control transition to next elements. Combination of multiple elements can realize a complex pulse sequence.
Exception	An operation control of a pulse generation channel is referred to as an “exception” here. • There are six exception types: Postpone, force, restart, transfer, start, and stop, each of which can be assigned with any event as its trigger.
Reference time division	Time division processing that delivers reference inputs to each pulse generation channel (0 to 15).
Reference slot	A unit of time that constitutes the reference time division (= pclk * 1).
Fixed event input	The event input used for the loop back of the PWM output. It can be used as a fixed input.
Selected event input	Event input of 16 channels selected from 64 channels, which are 58 channels of external event inputs and 6 channels of software event inputs. Event inputs that can be used by APA concurrently are 32 channels, which consists of fixed events (16 channels) and selected event inputs (16 channels).
Event time division	Event time division assigns time available for circuits in APA (pulse generation channels, P-Bus) for sharing hardware resources such as the element RAM and the calculation device.
Event slot	A unit of time that constitutes the event time division (= pclk * 1).
Shared cycle	The cycle of event time division. A cycle consists of 20 event slots (pclk * 20).

Table 22.5 List of Terms (2/2)

Terms	Definition in This Section
Pulse generation channel	A circuit that generates a PWM waveform. This type of circuit monitors a set of reference inputs to generate a PWM waveform that satisfies the conditions. Each module has 16 channels, each of which can operate independently.
ON match/OFF match	A match is a state, in which a reference input satisfies the pulse generating condition defined in an element, for generating PWM pulses on a pulse generation channel. A condition for enabling pulses is an ON match, and that for disabling pulses is an OFF match.
ON waiting/OFF waiting	ON waiting is a state in which ON match is being monitored. OFF waiting is a condition in which OFF match is being monitored. ON waiting and OFF waiting are mutually exclusive.
Absolute designation	Absolute designation is the specification of absolute thresholds for matching with references that turn outputs on and off. That is, in absolute designation, the thresholds are directly compared with reference values. The opposite of absolute is relative.
Relative designation	Relative designation is the specification of differences from the reference value at the time the threshold was specified. Outputs are turned on and off when the references reach these differences. The condition takes the form of a difference, so is not directly compared with the reference values (doing this requires converting the differences to absolute thresholds). The opposite of relative is absolute.
Time resolution	Minimum time unit (in terms of clock cycles) of a signal that can be input to or output from APA. For example, when pclk*4 is defined for an input signal (reference input A in **Table 22.1, Reference Input**), APA can process input signals whose period is 4 pclk cycles or more. When pclk*1 is defined for an output signal (PWM output in **Table 22.4, Pulse/Interrupt Output**), the signal will be toggled every pclk cycle.

22.3 Configuration

Figure 22.3 is a block diagram of APA.



Note 1. Reference generators are selectable by the APA input selector. See **Section 22.5.8, APA Input Selector**.

Figure 22.3 APA Block Diagram

Table 22.6 lists the event input sources.

Table 22.6 External Event Sources

External Event Source No.	Function	Remarks
External event sources 0 to 15	APA interrupt output n (m: 0 to 15)	
External event sources 16 to 19	ATU timer D0m compare match A (m: 0 to 3)	
External event sources 20 to 23	ATU timer D0m compare match B (m: 0 to 3)	
External event sources 24 to 27	ATU TOD0mA pin output (m: 0 to 3)	
External event sources 28 to 31	ATU TOD0mB pin output (m: 0 to 3)	
External event sources 32 to 35	ATU Timer D0m down counter underflow (m: 0 to 3)	
External event sources 36 to 43	Timer G compare match interrupt (channel m) (m: 0 to 7)	
External event source 44	2-bit error detection interrupt from the ECC for APA	Outputs of apa_ecc_0, apa_ecc_1, and apa_ecc_2 (1 bit each) are logically ORed to be one bit.
External event sources 45 to 48	Condition match interrupt of DFE channels 6 to 9	
External event sources 49 to 56	Not used	Low-Clamp
External event sources 57	Fixed value input for unused event processing	Low-Clamp See **Section 22.5.7.5, Handling of Unused Exceptions**.

22.4 Control Registers

22.4.1 List of Control Registers

APA registers are arranged on the P-Bus as shown in **Table 22.7**.

Register addresses are given as offsets from the base address FFEB 0000_H.

Table 22.7 Register Map (APA Local) (1/17)

+3	+2	+1	+0	Offset Address
000000_H			APAA0EN	0000
0000 0000_H				0004
APAA0CHEN				0008
0000_H		APAA0CHST		000C
0000 0000_H				0010 to 01FF
0000_H		APAA0RFDT0		0200
0000_H		APAA0RFDT1		0204
0000_H		APAA0RFDT2		0208
0000_H		APAA0RFDT3		020C
0000_H		APAA0RFDT4		0210
0000_H		APAA0RFDT5		0214
0000_H		APAA0RFDT6		0218
0000_H		APAA0RFDT7		021C
0000_H		APAA0RFDT8		0220
0000_H		APAA0RFDT9		0224
0000_H		APAA0RFSW		0228
0000 0000_H				022C
0000 0000_H				0230
0000 0000_H				0234
0000 0000_H				0238
0000 0000_H				023C
0000_H		APAA0RFMX0		0240
0000_H		APAA0RFMX1		0244
0000 0000_H				0248
0000 0000_H				024C
0000 0000_H				0250
0000 0000_H				0254
0000 0000_H				0258
0000 0000_H				025C
0000 0000_H				0260
0000 0000_H				0264
0000_H		APAA0RFMXSW		0268
0000 0000_H				026C
0000 0000_H				0270
0000 0000_H				0274
0000 0000_H				0278
0000 0000_H				027C
0000 0000_H				0280

Table 22.7 Register Map (APA Local) (2/17)

+3	+2	+1	+0	Offset Address
	000000_H		APAA0EVSL00	0400
	000000_H		APAA0EVSL01	0404
	000000_H		APAA0EVSL02	0408
	000000_H		APAA0EVSL03	040C
	000000_H		APAA0EVSL04	0410
	000000_H		APAA0EVS0L5	0414
	000000_H		APAA0EVSL06	0418
	000000_H		APAA0EVSL07	041C
	000000_H		APAA0EVSL08	0420
	000000_H		APAA0EVSL09	0424
	000000_H		APAA0EVSL10	0428
	000000_H		APAA0EVSL11	042C
	000000_H		APAA0EVSL12	0430
	000000_H		APAA0EVSL13	0434
	000000_H		APAA0EVSL14	0438
	000000_H		APAA0EVSL15	043C
	APAA0EVSW			0440
0000_H		APAA0EVSC		0444
	APAA0ESTA			0448
	$0000\ 0000_H$			044C
	$0000\ 0000_H$			0450
	$0000\ 0000_H$			0454
	APAA0CCGA00			0600
	APAA0CCGB00			0604
	APAA0CSTA00			0608
	APAA0CSTB00			060C
	APAA0CSTC00			0610
	APAA0CSTD00			0614
	$0000\ 0000_H$			0618
	0000 0000			061C
	APAA0CCGA01			0620
	APAA0CCGB01			0624
	APAA0CSTA01			0628
	APAA0CSTB01			062C
	APAA0CSTC01			0630
	APAA0CSTD01			0634
	$0000\ 0000_H$			0638
	$0000\ 0000_H$			063C
	APAA0CCGA02			0640
	APAA0CCGB02			0644
	APAA0CSTA02			0648
	APAA0CSTB02			064C
	APAA0CSTC02			0650
	APAA0CSTD02			0654

Table 22.7 Register Map (APA Local) (3/17)

+3	+2	+1	+0	Offset Address
	0000 0000_H			0658
	0000 0000_H			065C
	APAA0CCGA03			0660
	APAA0CCGB03			0664
	APAA0CSTA03			0668
	APAA0CSTB03			066C
	APAA0CSTC03			0670
	APAA0CSTD03			0674
	0000 0000_H			0678
	0000 0000_H			067C
	APAA0CCGA04			0680
	APAA0CCGB04			0684
	APAA0CSTA04			0688
	APAA0CSTB04			068C
	APAA0CSTC04			0690
	APAA0CSTD04			0694
	0000 0000_H			0698
	0000 0000_H			069C
	APAA0CCGA05			06A0
	APAA0CCGB05			06A4
	APAA0CSTA05			06A8
	APAA0CSTB05			06AC
	APAA0CSTC05			06B0
	APAA0CSTD05			06B4
	0000 0000_H			06B8
	0000 0000_H			06BC
	APAA0CCGA06			06C0
	APAA0CCGB06			06C4
	APAA0CSTA06			06C8
	APAA0CSTB06			06CC
	APAA0CSTC06			06D0
	APAA0CSTD06			06D4
	0000 0000_H			06D8
	0000 0000_H			06DC
	APAA0CCGA07			06E0
	APAA0CCGB07			06E4
	APAA0CSTA07			06E8
	APAA0CSTB07			06EC
	APAA0CSTC07			06F0
	APAA0CSTD07			06F4
	0000 0000_H			06F8
	0000 0000_H			06FC
	APAA0CCGA08			0700
	APAA0CCGB08			0704

Table 22.7 Register Map (APA Local) (4/17)

+3	+2	+1	+0	Offset Address
APAA0CSTA08				0708
APAA0CSTB08				070C
APAA0CSTC08				0710
APAA0CSTD08				0714
$0000\ 0000_H$				0718
$0000\ 0000_H$				071C
APAA0CCGA09				0720
APAA0CCGB09				0724
APAA0CSTA09				0728
APAA0CSTB09				072C
APAA0CSTC09				0730
APAA0CSTD09				0734
$0000\ 0000_H$				0738
$0000\ 0000_H$				073C
APAA0CCGA10				0740
APAA0CCGB10				0744
APAA0CSTA10				0748
APAA0CSTB10				074C
APAA0CSTC10				0750
APAA0CSTD10				0754
$0000\ 0000_H$				0758
$0000\ 0000_H$				075C
APAA0CCGA11				0760
APAA0CCGB11				0764
APAA0CSTA11				0768
APAA0CSTB11				076C
APAA0CSTC11				0770
APAA0CSTD11				0774
$0000\ 0000_H$				0778
$0000\ 0000_H$				077C
APAA0CCGA12				0780
APAA0CCGB12				0784
APAA0CSTA12				0788
APAA0CSTB12				078C
APAA0CSTC12				0790
APAA0CSTD12				0794
$0000\ 0000_H$				0798
$0000\ 0000_H$				079C
APAA0CCGA13				07A0
APAA0CCGB13				07A4
APAA0CSTA13				07A8
APAA0CSTB13				07AC
APAA0CSTC13				07B0
APAA0CSTD13				07B4

Table 22.7 Register Map (APA Local) (5/17)

+3	+2	+1	+0	Offset Address
		0000 0000_H		07B8
		0000 0000_H		07BC
		APAA0CCGA14		07C0
		APAA0CCGB14		07C4
		APAA0CSTA14		07C8
		APAA0CSTB14		07CC
		APAA0CSTC14		07D0
		APAA0CSTD14		07D4
		0000 0000_H		07D8
		0000 0000_H		07DC
		APAA0CCGA15		07E0
		APAA0CCGB15		07E4
		APAA0CSTA15		07E8
		APAA0CSTB15		07EC
		APAA0CSTC15		07F0
		APAA0CSTD15		07F4
		0000 0000_H		07F8
		0000 0000H		07FC
		APAA0ELMA000		1000
		APAA0ELMB000		1004
		APAA0ELMC000		1008
		0000 0000_H		100C
		APAA0ELMA001		1010
		APAA0ELMB001		1014
		APAA0ELMC001		1018
		0000 0000_H		101C
		APAA0ELMA002		1020
		APAA0ELMB002		1024
		APAA0ELMC002		1028
		0000 0000_H		102C
		APAA0ELMA003		1030
		APAA0ELMB003		1034
		APAA0ELMC003		1038
		0000 0000_H		103C
		APAA0ELMA004		1040
		APAA0ELMB004		1044
		APAA0ELMC004		1048
		0000 0000_H		104C
		APAA0ELMA005		1050
		APAA0ELMB005		1054
		APAA0ELMC005		1058
		0000 0000_H		105C
		APAA0ELMA006		1060
		APAA0ELMB006		1064

Table 22.7 Register Map (APA Local) (6/17)

+3	+2	+1	+0	Offset Address
APAA0ELMC006				1068
0000 0000_H				106C
APAA0ELMA007				1070
APAA0ELMB007				1074
APAA0ELMC007				1078
0000 0000_H				107C
APAA0ELMA008				1080
APAA0ELMB008				1084
APAA0ELMC008				1088
0000 0000_H				108C
APAA0ELMA009				1090
APAA0ELMB009				1094
APAA0ELMC009				1098
0000 0000_H				109C
APAA0ELMA010				10A0
APAA0ELMB010				10A4
APAA0ELMC010				10A8
0000 0000_H				10AC
APAA0ELMA011				10B0
APAA0ELMB011				10B4
APAA0ELMC011				10B8
0000 0000_H				10BC
APAA0ELMA012				10C0
APAA0ELMB012				10C4
APAA0ELMC012				10C8
0000 0000_H				10CC
APAA0ELMA013				10D0
APAA0ELMB013				10D4
APAA0ELMC013				10D8
0000 0000_H				10DC
APAA0ELMA014				10E0
APAA0ELMB014				10E4
APAA0ELMC014				10E8
0000 0000_H				10EC
APAA0ELMA015				10F0
APAA0ELMB015				10F4
APAA0ELMC015				10F8
0000 0000_H				10FC
APAA0ELMA016				1100
APAA0ELMB016				1104
APAA0ELMC016				1108
0000 0000_H				110C
APAA0ELMA017				1110
APAA0ELMB017				1114

Table 22.7 Register Map (APA Local) (7/17)

+3	+2	+1	+0	Offset Address
APAA0ELMC017				1118
0000 0000_H				111C
APAA0ELMA018				1120
APAA0ELMB018				1124
APAA0ELMC018				1128
0000 0000_H				112C
APAA0ELMA019				1130
APAA0ELMB019				1134
APAA0ELMC019				1138
0000 0000_H				113C
APAA0ELMA020				1140
APAA0ELMB020				1144
APAA0ELMC020				1148
0000 0000_H				114C
APAA0ELMA021				1150
APAA0ELMB021				1154
APAA0ELMC021				1158
0000 0000_H				115C
APAA0ELMA022				1160
APAA0ELMB022				1164
APAA0ELMC022				1168
0000 0000_H				116C
APAA0ELMA023				1170
APAA0ELMB023				1174
APAA0ELMC023				1178
0000 0000_H				117C
APAA0ELMA024				1180
APAA0ELMB024				1184
APAA0ELMC024				1188
0000 0000_H				118C
APAA0ELMA025				1190
APAA0ELMB025				1194
APAA0ELMC025				1198
0000 0000_H				119C
APAA0ELMA026				11A0
APAA0ELMB026				11A4
APAA0ELMC026				11A8
0000 0000_H				11AC
APAA0ELMA027				11B0
APAA0ELMB027				11B4
APAA0ELMC027				11B8
0000 0000_H				11BC
APAA0ELMA028				11C0
APAA0ELMB028				11C4

Table 22.7 Register Map (APA Local) (8/17)

+3	+2	+1	+0	Offset Address
APAA0ELMC028				11C8
$0000\ 0000_H$				11CC
APAA0ELMA029				11D0
APAA0ELMB029				11D4
APAA0ELMC029				11D8
$0000\ 0000_H$				11DC
APAA0ELMA030				11E0
APAA0ELMB030				11E4
APAA0ELMC030				11E8
$0000\ 0000_H$				11EC
APAA0ELMA031				11F0
APAA0ELMB031				11F4
APAA0ELMC031				11F8
$0000\ 0000_H$				11FC
APAA0ELMA032				1200
APAA0ELMB032				1204
APAA0ELMC032				1208
$0000\ 0000_H$				120C
APAA0ELMA033				1210
APAA0ELMB033				1214
APAA0ELMC033				1218
$0000\ 0000_H$				121C
APAA0ELMA034				1220
APAA0ELMB034				1224
APAA0ELMC034				1228
$0000\ 0000_H$				122C
APAA0ELMA035				1230
APAA0ELMB035				1234
APAA0ELMC035				1238
$0000\ 0000_H$				123C
APAA0ELMA036				1240
APAA0ELMB036				1244
APAA0ELMC036				1248
$0000\ 0000_H$				124C
APAA0ELMA037				1250
APAA0ELMB037				1254
APAA0ELMC037				1258
$0000\ 0000_H$				125C
APAA0ELMA038				1260
APAA0ELMB038				1264
APAA0ELMC038				1268
$0000\ 0000_H$				126C
APAA0ELMA039				1270
APAA0ELMB039				1274

Table 22.7 Register Map (APA Local) (9/17)

+3	+2	+1	+0	Offset Address
APAA0ELMC039				1278
0000 0000_H				127C
APAA0ELMA040				1280
APAA0ELMB040				1284
APAA0ELMC040				1288
0000 0000_H				128C
APAA0ELMA041				1290
APAA0ELMB041				1294
APAA0ELMC041				1298
0000 0000_H				129C
APAA0ELMA042				12A0
APAA0ELMB042				12A4
APAA0ELMC042				12A8
0000 0000_H				12AC
APAA0ELMA043				12B0
APAA0ELMB043				12B4
APAA0ELMC043				12B8
0000 0000_H				12BC
APAA0ELMA044				12C0
APAA0ELMB044				12C4
APAA0ELMC044				12C8
0000 0000_H				12CC
APAA0ELMA045				12D0
APAA0ELMB045				12D4
APAA0ELMC045				12D8
0000 0000_H				12DC
APAA0ELMA046				12E0
APAA0ELMB046				12E4
APAA0ELMC046				12E8
0000 0000_H				12EC
APAA0ELMA047				12F0
APAA0ELMB047				12F4
APAA0ELMC047				12F8
0000 0000_H				12FC
APAA0ELMA048				1300
APAA0ELMB048				1304
APAA0ELMC048				1308
0000 0000_H				130C
APAA0ELMA049				1310
APAA0ELMB049				1314
APAA0ELMC049				1318
0000 0000_H				131C
APAA0ELMA050				1320
APAA0ELMB050				1324

Table 22.7 Register Map (APA Local) (10/17)

+3	+2	+1	+0	Offset Address
APAA0ELMC050				1328
0000 0000_H				132C
APAA0ELMA051				1330
APAA0ELMB051				1334
APAA0ELMC051				1338
0000 0000_H				133C
APAA0ELMA052				1340
APAA0ELMB052				1344
APAA0ELMC052				1348
0000 0000_H				134C
APAA0ELMA053				1350
APAA0ELMB053				1354
APAA0ELMC053				1358
0000 0000_H				135C
APAA0ELMA054				1360
APAA0ELMB054				1364
APAA0ELMC054				1368
0000 0000_H				136C
APAA0ELMA055				1370
APAA0ELMB055				1374
APAA0ELMC055				1378
0000 0000_H				137C
APAA0ELMA056				1380
APAA0ELMB056				1384
APAA0ELMC056				1388
0000 0000_H				138C
APAA0ELMA057				1390
APAA0ELMB057				1394
APAA0ELMC057				1398
0000 0000_H				139C
APAA0ELMA058				13A0
APAA0ELMB058				13A4
APAA0ELMC058				13A8
0000 0000_H				13AC
APAA0ELMA059				13B0
APAA0ELMB059				13B4
APAA0ELMC059				13B8
0000 0000_H				13BC
APAA0ELMA060				13C0
APAA0ELMB060				13C4
APAA0ELMC060				13C8
0000 0000_H				13CC
APAA0ELMA061				13D0
APAA0ELMB061				13D4

Table 22.7 Register Map (APA Local) (11/17)

+3	+2	+1	+0	Offset Address
APAA0ELMC061				13D8
$0000\ 0000_H$				13DC
APAA0ELMA062				13E0
APAA0ELMB062				13E4
APAA0ELMC062				13E8
$0000\ 0000_H$				13EC
APAA0ELMA063				13F0
APAA0ELMB063				13F4
APAA0ELMC063				13F8
$0000\ 0000_H$				13FC
APAA0ELMA064				1400
APAA0ELMB064				1404
APAA0ELMC064				1408
$0000\ 0000_H$				140C
APAA0ELMA065				1410
APAA0ELMB065				1414
APAA0ELMC065				1418
$0000\ 0000_H$				141C
APAA0ELMA066				1420
APAA0ELMB066				1424
APAA0ELMC066				1428
$0000\ 0000_H$				142C
APAA0ELMA067				1430
APAA0ELMB067				1434
APAA0ELMC067				1438
$0000\ 0000_H$				143C
APAA0ELMA068				1440
APAA0ELMB068				1444
APAA0ELMC068				1448
$0000\ 0000_H$				144C
APAA0ELMA069				1450
APAA0ELMB069				1454
APAA0ELMC069				1458
$0000\ 0000_H$				145C
APAA0ELMA070				1460
APAA0ELMB070				1464
APAA0ELMC070				1468
$0000\ 0000_H$				146C
APAA0ELMA071				1470
APAA0ELMB071				1474
APAA0ELMC071				1478
$0000\ 0000_H$				147C
APAA0ELMA072				1480
APAA0ELMB072				1484

Table 22.7 Register Map (APA Local) (12/17)

+3	+2	+1	+0	Offset Address
APAA0ELMC072				1488
0000 0000_H				148C
APAA0ELMA073				1490
APAA0ELMB073				1494
APAA0ELMC073				1498
0000 0000_H				149C
APAA0ELMA074				14A0
APAA0ELMB074				14A4
APAA0ELMC074				14A8
0000 0000_H				14AC
APAA0ELMA075				14B0
APAA0ELMB075				14B4
APAA0ELMC075				14B8
0000 0000_H				14BC
APAA0ELMA076				14C0
APAA0ELMB076				14C4
APAA0ELMC076				14C8
0000 0000_H				14CC
APAA0ELMA077				14D0
APAA0ELMB077				14D4
APAA0ELMC077				14D8
0000 0000_H				14DC
APAA0ELMA078				14E0
APAA0ELMB078				14E4
APAA0ELMC078				14E8
0000 0000_H				14EC
APAA0ELMA079				14F0
APAA0ELMB079				14F4
APAA0ELMC079				14F8
0000 0000_H				14FC
APAA0ELMA080				1500
APAA0ELMB080				1504
APAA0ELMC080				1508
0000 0000_H				150C
APAA0ELMA081				1510
APAA0ELMB081				1514
APAA0ELMC081				1518
0000 0000_H				151C
APAA0ELMA082				1520
APAA0ELMB082				1524
APAA0ELMC082				1528
0000 0000_H				152C
APAA0ELMA083				1530
APAA0ELMB083				1534

Table 22.7 Register Map (APA Local) (13/17)

+3	+2	+1	+0	Offset Address
APAA0ELMC083				1538
0000 0000_H				153C
APAA0ELMA084				1540
APAA0ELMB084				1544
APAA0ELMC084				1548
0000 0000_H				154C
APAA0ELMA085				1550
APAA0ELMB085				1554
APAA0ELMC085				1558
0000 0000_H				155C
APAA0ELMA086				1560
APAA0ELMB086				1564
APAA0ELMC086				1568
0000 0000_H				156C
APAA0ELMA087				1570
APAA0ELMB087				1574
APAA0ELMC087				1578
0000 0000_H				157C
APAA0ELMA088				1580
APAA0ELMB088				1584
APAA0ELMC088				1588
0000 0000_H				158C
APAA0ELMA089				1590
APAA0ELMB089				1594
APAA0ELMC089				1598
0000 0000_H				159C
APAA0ELMA090				15A0
APAA0ELMB090				15A4
APAA0ELMC090				15A8
0000 0000_H				15AC
APAA0ELMA091				15B0
APAA0ELMB091				15B4
APAA0ELMC091				15B8
0000 0000_H				15BC
APAA0ELMA092				15C0
APAA0ELMB092				15C4
APAA0ELMC092				15C8
0000 0000_H				15CC
APAA0ELMA093				15D0
APAA0ELMB093				15D4
APAA0ELMC093				15D8
0000 0000_H				15DC
APAA0ELMA094				15E0
APAA0ELMB094				15E4

Table 22.7 Register Map (APA Local) (14/17)

+3	+2	+1	+0	Offset Address
APAA0ELMC094				15E8
0000 0000_H				15EC
APAA0ELMA095				15F0
APAA0ELMB095				15F4
APAA0ELMC095				15F8
0000 0000_H				15FC
APAA0ELMA096				1600
APAA0ELMB096				1604
APAA0ELMC096				1608
0000 0000_H				160C
APAA0ELMA097				1610
APAA0ELMB097				1614
APAA0ELMC097				1618
0000 0000_H				161C
APAA0ELMA098				1620
APAA0ELMB098				1624
APAA0ELMC098				1628
0000 0000_H				162C
APAA0ELMA099				1630
APAA0ELMB099				1634
APAA0ELMC099				1638
0000 0000_H				163C
APAA0ELMA100				1640
APAA0ELMB100				1644
APAA0ELMC100				1648
0000 0000_H				164C
APAA0ELMA101				1650
APAA0ELMB101				1654
APAA0ELMC101				1658
0000 0000_H				165C
APAA0ELMA102				1660
APAA0ELMB102				1664
APAA0ELMC102				1668
0000 0000_H				166C
APAA0ELMA103				1670
APAA0ELMB103				1674
APAA0ELMC103				1678
0000 0000_H				167C
APAA0ELMA104				1680
APAA0ELMB104				1684
APAA0ELMC104				1688
0000 0000_H				168C
APAA0ELMA105				1690
APAA0ELMB105				1694

Table 22.7 Register Map (APA Local) (15/17)

+3	+2	+1	+0	Offset Address
APAA0ELMC105				1698
$0000\ 0000_H$				169C
APAA0ELMA106				16A0
APAA0ELMB106				16A4
APAA0ELMC106				16A8
$0000\ 0000_H$				16AC
APAA0ELMA107				16B0
APAA0ELMB107				16B4
APAA0ELMC107				16B8
$0000\ 0000_H$				16BC
APAA0ELMA108				16C0
APAA0ELMB108				16C4
APAA0ELMC108				16C8
$0000\ 0000_H$				16CC
APAA0ELMA109				16D0
APAA0ELMB109				16D4
APAA0ELMC109				16D8
$0000\ 0000_H$				16DC
APAA0ELMA110				16E0
APAA0ELMB110				16E4
APAA0ELMC110				16E8
$0000\ 0000_H$				16EC
APAA0ELMA111				16F0
APAA0ELMB111				16F4
APAA0ELMC111				16F8
$0000\ 0000_H$				16FC
APAA0ELMA112				1700
APAA0ELMB112				1704
APAA0ELMC112				1708
$0000\ 0000_H$				170C
APAA0ELMA113				1710
APAA0ELMB113				1714
APAA0ELMC113				1718
$0000\ 0000_H$				171C
APAA0ELMA114				1720
APAA0ELMB114				1724
APAA0ELMC114				1728
$0000\ 0000_H$				172C
APAA0ELMA115				1730
APAA0ELMB115				1734
APAA0ELMC115				1738
$0000\ 0000_H$				173C
APAA0ELMA116				1740
APAA0ELMB116				1744

Table 22.7 Register Map (APA Local) (16/17)

+3	+2	+1	+0	Offset Address
APAA0ELMC116				1748
0000 0000_H				174C
APAA0ELMA117				1750
APAA0ELMB117				1754
APAA0ELMC117				1758
0000 0000_H				175C
APAA0ELMA118				1760
APAA0ELMB118				1764
APAA0ELMC118				1768
0000 0000_H				176C
APAA0ELMA119				1770
APAA0ELMB119				1774
APAA0ELMC119				1778
0000 0000_H				177C
APAA0ELMA120				1780
APAA0ELMB120				1784
APAA0ELMC120				1788
0000 0000_H				178C
APAA0ELMA121				1790
APAA0ELMB121				1794
APAA0ELMC121				1798
0000 0000_H				179C
APAA0ELMA122				17A0
APAA0ELMB122				17A4
APAA0ELMC122				17A8
0000 0000_H				17AC
APAA0ELMA123				17B0
APAA0ELMB123				17B4
APAA0ELMC123				17B8
0000 0000_H				17BC
APAA0ELMA124				17C0
APAA0ELMB124				17C4
APAA0ELMC124				17C8
0000 0000_H				17CC
APAA0ELMA125				17D0
APAA0ELMB125				17D4
APAA0ELMC125				17D8
0000 0000_H				17DC
APAA0ELMA126				17E0
APAA0ELMB126				17E4
APAA0ELMC126				17E8
0000 0000_H				17EC

Table 22.7 Register Map (APA Local) (17/17)

+3	+2	+1	+0	Offset Address
		APAA0ELMA127		17F0
		APAA0ELMB127		17F4
		APAA0ELMC127		17F8

Remarks: 17FC to 1FFC are reserved.

22.4.2 Control Register Details (Overall Operation)

Registers explained here configure the overall operation of APA.

Table 22.8 List of Control Registers (Overall Operation)

Offset Address	Symbol	Register Name	Value after Reset	Access Size	Function
0000_H	APAA0EN	APAA0 operation enable register	00_H	8	Enabling APA overall operation
0008_H	APAA0CHEN	APAA0 channel operation enable register	0000_0000_H	32	Enabling all pulse generation channels
$000C_H$	APAA0CHST	APAA0 channel output status register	0000_H	16	Operation status of all pulse generation channels (PWM)
0010_H to $01FF_H$	(N/A)		0000_0000_H	—	Unused area (read-only) **Note:** This area is read as 0 and cannot be modified.

22.4.2.1 APAA0EN — APAA0 Operation Enable Register

This register controls starts and stops of the reference bus and the event bus of APA.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	APAACMENAPAA
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 22.9 APAA0EN Register Contents

Bit Position	Bit Name	Reset Cause	Function
0	APAACMENAPA	PRESETZ	Enabling APA operation. 0: Disabling stops APA 1: Enabling starts APA

Table 22.10 Setting to Enable Each Function

		Function		
Setting Register	Enable Setting	Reference Control	Event Control	Pulse generation channel (0 to 15)
APAA0EN	Writing 1 though the P-Bus	√	√	—
APAA0CHEN[15:0]	Writing 1 though the P-Bus. The setting can be made independently for channels 0 to 15.	—	—	√

NOTES

1. Change of this register must be made while all pulse generation channels (16 channels) are stopped (APAA0CHEN register = 0000 0000_H).
2. Pulse generation channels are not initialized even if this register is cleared to 0. Thus, if APAACMENAPA = low is set before pulse generation channel operation is stopped, unintended PWM waveforms may be output because the pulse generation channel stops with the values retained.
3. For the settings to start and stop operation, see **Section 22.5.7.4, Operation Procedure**.
4. Enabling APA starts reference control and event control operation.

22.4.2.2 APAA0CHEN — APAA0 Channel Operation Enable Register

This register enables all pulse generation channels. Each bit of the APAA0CHEN[15:0] register controls ON/OFF of a pulse generation channel.

Each bit of the APAA0CHEN[15:0] register can be written to only if the corresponding bit of the mask register (APAACMENMSKPGCH[15:0]) is set to 1.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	APAACMENMSKPGCH[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	APAACMENPGCH15	APAACMENPGCH14	APAACMENPGCH13	APAACMENPGCH12	APAACMENPGCH11	APAACMENPGCH10	APAACMENPGCH9	APAACMENPGCH8	APAACMENPGCH7	APAACMENPGCH6	APAACMENPGCH5	APAACMENPGCH4	APAACMENPGCH3	APAACMENPGCH2	APAACMENPGCH1	APAACMENPGCH0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 22.11 APAA0CHEN Register Contents

Bit Position	Bit Name	Reset Cause	Function
31 to 16	APAACMENMSKPGCH[15:0]	—	Mask writing to APAACMENPGCH[15:0]. Each bit is assigned for control of PGCH15 to PGCH0. 0: Writing to APAACMENPGCH[n] is disabled. 1: Writing to APAACMENPGCH[n] is enabled. **Note:** These bits are read as 0.
15 to 0	APAACMENPGCH[15:0]	PRESETZ	Set On/Off of PGCH. Each bit is assigned for control of PGCH15 to PGCH0. 0: Operation of PGCH15 to PGCH0 is disabled. 1: Operation of PGCH15 to PGCH0 is enabled.

The following examples show the settings to enable or disable the operation of the pulse generation channels (PGCHs).

- Writing 00FF $FFFF_H$: Operation of PGCH7 to PGCH0 is enabled.
- Writing 0000 $00FF_H$: No effect (The registers retain the previous values.)
- Writing FFFF 0000_H: Operation of PGCH15 to PGCH0 is disabled.
- Writing 0000 0000_H: No effect (The registers retain the previous values.)

Table 22.12 shows the correspondence between the bits of APAACMENMSKPGCH[15:0] and APAACMENPGCH[15:0] and PGCHs.

Table 22.12 Bit Correspondence to PGCH

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Corresponding channels	PGCH15	PGCH14	PGCH13	PGCH12	PGCH11	PGCH10	PGCH9	PGCH8	PGCH7	PGCH6	PGCH5	PGCH4	PGCH3	PGCH2	PGCH1	PGCH0

NOTES

1. Event time division must be in operation (the APAA0EN register = 1) when this register is set to 1 with pulse generation channels stopped. If this register is modified when event time division is not in operation (the APAA0EN register = 0), the intended matching comparison is not performed because the reference bus and the event bus do not work.
2. If APAACMENPGCH[n] is changed from 1 to 0 (channel operation enabled → disabled), one or two shared cycles of delay are inserted before the operation is stopped.
3. For the settings to start and stop operation, see **Section 22.5.7.4, Operation Procedure**.
4. For the pin state with APAACMENPGCH[n] = low, see **(5) Channel ON/OFF** in **Section 22.5.4.2, Operation: Matching Comparator**.

22.4.2.3 APAA0CHST — APAA0 Channel Output Status Register

This register monitors the operation status of all pulse generation channels (PWM). Each bit indicates level of the current PWM output pin (APAAEOPWM[15:0]) of each pulse generation channel.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	APAACMSTATPWM15	APAACMSTATPWM14	APAACMSTATPWM13	APAACMSTATPWM12	APAACMSTATPWM11	APAACMSTATPWM10	APAACMSTATPWM9	APAACMSTATPWM8	APAACMSTATPWM7	APAACMSTATPWM6	APAACMSTATPWM5	APAACMSTATPWM4	APAACMSTATPWM3	APAACMSTATPWM2	APAACMSTATPWM1	APAACMSTATPWM0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 22.13 APAA0CHST Register Contents

Bit Position	Bit Name	Reset Cause	Function
15 to 0	APAACMSTATPWM [15:0]	PRESETZ	Indicate the PWM output level of each pulse generation channel (0 to 15). For the correspondence between each bit and PGCH, see **Table 22.12**.

22.4.3 Control Register Details (Reference Bus)

These registers configure and read information about the operation of the reference bus.

Table 22.14 List of Control Registers (Reference Bus)

Offset Address	Symbol	Register Name	Value after Reset	Access Size	Function
0200_H	APAA0RFDT0	APAA0 reference data register 0	0000_H	16	Reads the value of reference input 0.
0204_H	APAA0RFDT1	APAA0 reference data register 1	0000_H	16	Reads the value of reference input 1.
0208_H	APAA0RFDT2	APAA0 reference data register 2	0000_H	16	Reads the value of reference input 2.
$020C_H$	APAA0RFDT3	APAA0 reference data register 3	0000_H	16	Reads the value of reference input 3.
0210_H	APAA0RFDT4	APAA0 reference data register 4	0000_H	16	Reads the value of reference input 4.
0214_H	APAA0RFDT5	APAA0 reference data register 5	0000_H	16	Reads the value of reference input 5.
0218_H	APAA0RFDT6	APAA0 reference data register 6	0000_H	16	Reads the value of reference input 6.
$021C_H$	APAA0RFDT7	APAA0 reference data register 7	0000_H	16	Reads the value of reference input 7.
0220_H	APAA0RFDT8	APAA0 reference data register 8	0000_H	16	Reads the value of reference input 8.
0224_H	APAA0RFDT9	APAA0 reference data register 9	0000_H	16	Reads the value of reference input 9.
0228_H	APAA0RFSW	APAA0 software reference data register	0000_H	16	Configures the value of the software reference input.
$022C_H$ to $023C_H$	(N/A)		$0000\ 0000_H$	—	Unused area (read-only) **Note:** This area is read as 0 and cannot be modified.
0240_H	APAA0RFMX0	APAA0 reference maximum value setting register 0	$FFFF_H$	16	Sets the maximum value of reference input 0.
0244_H	APAA0RFMX1	APAA0 reference maximum value setting register 1	$FFFF_H$	16	Sets the maximum value of reference input 1.
0248_H to 0264_H	(N/A)		$0000\ 0000_H$	—	Unused area (read-only) **Note:** This area is read as 0 and cannot be modified.
0268_H	APAA0RFMXSW	APAA0 software reference maximum value setting register	$FFFF_H$	16	Sets the maximum value of the software reference input.
$026C_H$ to $03FC_H$	(N/A)		$0000\ 0000_H$	—	Unused area (read-only) **Note:** This area is read as 0 and cannot be modified.

22.4.3.1 APAA0RFDTn — APAA0 Reference Data Registers (n = 0 to 9)

These registers read the value of reference inputs. The value to be read is the latest value that is output on the reference bus.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	APAAREFBDAT[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 22.15 APAA0RFDTn Register Contents

Bit Position	Bit Name	Reset Cause	Function
15 to 0	APAAREFBDAT[15:0]	PRESETZ	Read the latest data of reference input n. **Note:** This register is effective only when the reference bus is effective (APAACMENAPA = 1). The current value is retained when the reference bus is disabled (APAACMENAPA = 0).

Table 22.16 shows the correspondence between registers and reference inputs.

Table 22.16 Correspondence between APAA0RFDTn Registers and Reference Inputs

		Reference Input
Register	Offset Address	Reference ID
APAA0RFDT0	0200_H	0
APAA0RFDT1	0204_H	1
APAA0RFDT2	0208_H	2
APAA0RFDT3	$020C_H$	3
APAA0RFDT4	0210_H	4
APAA0RFDT5	0214_H	5
APAA0RFDT6	0218_H	6
APAA0RFDT7	$021C_H$	7
APAA0RFDT8	0220_H	8
APAA0RFDT9	0224_H	9

22.4.3.2 APAA0RFSW — APAA0 Software Reference Data Register

This register configures and reads the value of the software reference input.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	APAAREFBDATSW[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 22.17 APAA0RFSW Register Contents

Bit Position	Bit Name	Reset Cause	Function
15 to 0	APAAREFBDATSW [15:0]	PRESETZ	Software reference input. It is specified as an unsigned value. The set value in this register becomes effective as reference input immediately after completion of writing. However, the setting of this register is reflected in a reference bus output (refb_dat_r[15:0]) when the reference slot is 12_H (see **Section 22.5.2.2, Operation**).

22.4.3.3 APAA0RFMXn — APAA0 Reference Maximum Value Setting Registers (n = 0, 1, sw)

These registers set the maximum values of reference inputs.

These registers are referenced when relative designation is employed for matching conditions (used for overflow processing that might occur when relative designation is converted to absolute designation). Relative destination is available for only references 0, 1, and sw. Absolute destination is only available for references 2 to 9.

CAUTION

When these registers are configured, all pulse generation channels must be stopped (APAACMENPGCH = 0000 0000_H). If any of these registers is configured in other conditions, matching comparison might operate in an unintended manner.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	APAAREFBMAX[15:0]															
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 22.18 APAA0RFMXn Register Contents

Bit Position	Bit Name	Reset Cause	Function
15 to 0	APAAREFBMAX[15:0]	PRESETZ	Set the maximum value of the software reference input. It is specified as an unsigned value.

Table 22.19 shows the correspondence between registers and reference inputs.

Table 22.19 Correspondence between APAA0RFMXn Registers and Reference Inputs

		Reference Input	
Register	**Offset Address**	**Reference ID**	**Time Resolution**
APAA0RFMX0	0240_H	0	pclk*4
APAA0RFMX1	0244_H	1	pclk*4
APAA0RFMXSW	0268_H	f	pclk*20

If the corresponding reference input is smaller than the dynamic range (16 bits), set any of these registers to a value that the corresponding reference input can take (dynamic range).

If the dynamic range of the reference input and the register value do not coincide, matching might not be obtained in the intended manner when relative designation is employed for matching comparison.

If the source counter value of the corresponding reference input is larger than the dynamic range of APA, care must be taken on the relationship with the reference input value of APA at an overflow of the source counter.

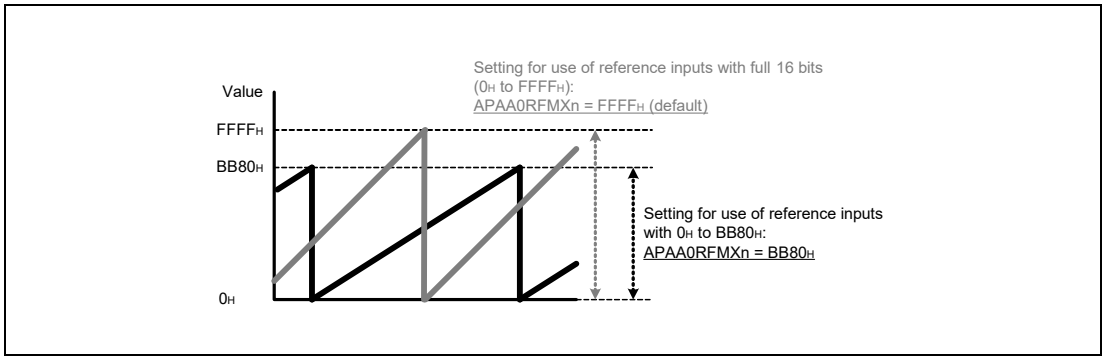


Figure 22.4 Configuration Example of APAA0RFMXn (n = 0/1/sw)

22.4.4 Control Register Details (Event Bus)

These registers configure and read information about the operation of the event buses.

Table 22.20 List of Control Registers (Event)

Offset Address	Symbol	Register Name	Value after Reset	Access Size	Function
0400_H	APAA0EVSL00	APAA0 Event Select Register 0	00_H	8	Setting of selected event input 0
0404_H	APAA0EVSL01	APAA0 Event Select Register 1	00_H	8	Setting of selected event input 1
0408_H	APAA0EVSL02	APAA0 Event Select Register 2	00_H	8	Setting of selected event input 2
$040C_H$	APAA0EVSL03	APAA0 Event Select Register 3	00_H	8	Setting of selected event input 3
0410_H	APAA0EVSL04	APAA0 Event Select Register 4	00_H	8	Setting of selected event input 4
0414_H	APAA0EVSL05	APAA0 Event Select Register 5	00_H	8	Setting of selected event input 5
0418_H	APAA0EVSL06	APAA0 Event Select Register 6	00_H	8	Setting of selected event input 6
$041C_H$	APAA0EVSL07	APAA0 Event Select Register 7	00_H	8	Setting of selected event input 7
0420_H	APAA0EVSL08	APAA0 Event Select Register 8	00_H	8	Setting of selected event input 8
0424_H	APAA0EVSL09	APAA0 Event Select Register 9	00_H	8	Setting of selected event input 9
0428_H	APAA0EVSL10	APAA0 Event Select Register 10	00_H	8	Setting of selected event input 10
$042C_H$	APAA0EVSL11	APAA0 Event Select Register 11	00_H	8	Setting of selected event input 11
0430_H	APAA0EVSL12	APAA0 Event Select Register 12	00_H	8	Setting of selected event input 12
0434_H	APAA0EVSL13	APAA0 Event Select Register 13	00_H	8	Setting of selected event input 13
0438_H	APAA0EVSL14	APAA0 Event Select Register 14	00_H	8	Setting of selected event input 14
$043C_H$	APAA0EVSL15	APAA0 Event Select Register 15	00_H	8	Setting of selected event input 15
0440_H	APAA0EVSW	APAA0 Software Event Register	$0000\ 0000_H$	32	Setting of software event
0444_H	APAA0EVSC	APAA0 Software Event Setting Register	0000_H	16	Setting of operation option of software event
0448_H	APAA0ESTA	APAA0 Event Status Register A	$0000\ 0000_H$	32	Operation status (level) of event bus
$044C_H$ to $05FF_H$	(N/A)		$0000\ 0000_H$		Unused area (read-only) **Note:** This area is read as 0 and cannot be modified.

22.4.4.1 APAA0EVSLn — APAA0 Event Select Registers (n = 00 to 15)

These registers select event inputs.

In the event bus, any 16 event inputs can be selected from among 64 channels, which consist of external event sources and software events 5 to 0. These registers configure the selection of event inputs.

Bit	7	6	5	4	3	2	1	0
	—	—	APAAEVSEL[5:0]					
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Table 22.21 APAA0EVSLn Register Contents

Bit Position	Bit Name	Reset Cause	Function
5 to 0	APAAEVSEL[5:0]	PRESETZ	Selects an event input. See **Table 22.22.**

CAUTION

When these registers are configured, all pulse generation channels must be stopped (APAACMENPGCH = 0000 0000_H). If any of these registers are configured in other conditions, event detection might operate in a wrong manner.

Table 22.22 shows the correspondence between the setting of APAAEVSEL[5:0] and the event input selection. Events that can be selected by each of APAAEVSEL[5:0] are called selected event inputs. For the correspondence between the setting of APAAEVSEL[5:0] and the event buses within the APA, see **Table 22.23**.

Table 22.22 Correspondence between APAAEVSEL[5:0] Setting and Event Inputs

Setting Value	Selected External Event Source	Setting Value	Selected External Event Source	Setting Value	Selected External Event Source	Setting Value	Selected External Event Source
00_H	External event source 0	10_H	External event source 16	20_H	External event source 32	30_H	External event source 48
01_H	External event source 1	11_H	External event source 17	21_H	External event source 33	31_H	External event source 49
02_H	External event source 2	12_H	External event source 18	22_H	External event source 34	32_H	External event source 50
03_H	External event source 3	13_H	External event source 19	23_H	External event source 35	33_H	External event source 51
04_H	External event source 4	14_H	External event source 20	24_H	External event source 36	34_H	External event source 52
05_H	External event source 5	15_H	External event source 21	25_H	External event source 37	35_H	External event source 53
06_H	External event source 6	16_H	External event source 22	26_H	External event source 38	36_H	External event source 54
07_H	External event source 7	17_H	External event source 23	27_H	External event source 39	37_H	External event source 55
08_H	External event source 8	18_H	External event source 24	28_H	External event source 40	38_H	External event source 56
09_H	External event source 9	19_H	External event source 25	29_H	External event source 41	39_H	External event source 57
$0A_H$	External event source 10	$1A_H$	External event source 26	$2A_H$	External event source 42	$3A_H$	Software event 0
$0B_H$	External event source 11	$1B_H$	External event source 27	$2B_H$	External event source 43	$3B_H$	Software event 1
$0C_H$	External event source 12	$1C_H$	External event source 28	$2C_H$	External event source 44	$3C_H$	Software event 2
$0D_H$	External event source 13	$1D_H$	External event source 29	$2D_H$	External event source 45	$3D_H$	Software event 3
$0E_H$	External event source 14	$1E_H$	External event source 30	$2E_H$	External event source 46	$3E_H$	Software event 4
$0F_H$	External event source 15	$1F_H$	External event source 31	$2F_H$	External event source 47	$3F_H$	Software event 5

The internal event buses eb_dat_r[31:0] have the following correspondence.

- eb_dat_r[15:0]: Correspond to PWM outputs n (n = 0 to 15) one by one (fixed event inputs).
- eb_dat_r[31:16]: Event inputs selected with the APAA0EVSL00 to APAA0EVSL 15 settings can be used (selected event inputs).

Table 22.23 Input Sources of Event Bus Signals

Fixed Event Input			Selected Event Input		
Event Bus	**Setting Register**	**Corresponding Pin**	**Event Bus**	**Setting Register**	**Offset Address**
eb_dat_r[0]	None	PWM output 0	eb_dat_r[16]	APAA0EVSL00[5:0]	0400_H
eb_dat_r[1]	None	PWM output 1	eb_dat_r[17]	APAA0EVSL01[5:0]	0404_H
eb_dat_r[2]	None	PWM output 2	eb_dat_r[18]	APAA0EVSL02[5:0]	0408_H
eb_dat_r[3]	None	PWM output 3	eb_dat_r[19]	APAA0EVSL03[5:0]	$040C_H$
eb_dat_r[4]	None	PWM output 4	eb_dat_r[20]	APAA0EVSL04[5:0]	0410_H
eb_dat_r[5]	None	PWM output 5	eb_dat_r[21]	APAA0EVSL05[5:0]	0414_H
eb_dat_r[6]	None	PWM output 6	eb_dat_r[22]	APAA0EVSL06[5:0]	0418_H
eb_dat_r[7]	None	PWM output 7	eb_dat_r[23]	APAA0EVSL07[5:0]	$041C_H$
eb_dat_r[8]	None	PWM output 8	eb_dat_r[24]	APAA0EVSL08[5:0]	0420_H
eb_dat_r[9]	None	PWM output 9	eb_dat_r[25]	APAA0EVSL09[5:0]	0424_H
eb_dat_r[10]	None	PWM output 10	eb_dat_r[26]	APAA0EVSL10[5:0]	0428_H
eb_dat_r[11]	None	PWM output 11	eb_dat_r[27]	APAA0EVSL11[5:0]	$042C_H$
eb_dat_r[12]	None	PWM output 12	eb_dat_r[28]	APAA0EVSL12[5:0]	0430_H
eb_dat_r[13]	None	PWM output 13	eb_dat_r[29]	APAA0EVSL13[5:0]	0434_H
eb_dat_r[14]	None	PWM output 14	eb_dat_r[30]	APAA0EVSL14[5:0]	0438_H
eb_dat_r[15]	None	PWM output 15	eb_dat_r[31]	APAA0EVSL15[5:0]	$043C_H$

22.4.4.2 APAA0EVSW — APAA0 Software Event Register

This register is the software event register. The value of each of bits APAAEVSW5 to APAAEVSW0 in this register can be used as an event input as it is. Each of bits APAAEVSW5 to APAAEVSW0 can be modified only if the corresponding mask bit (APAAEVSWMSK5 to APAAEVSWMSK0) is set to 1.

The operation option can be set by bits APAAEVSWCFG5 to APAAEVSWCFG0 in the APAA0EVSC register.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	APAAEVSWMSK[5:0]					
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	APAAEVSW5	APAAEVSW4	APAAEVSW3	APAAEVSW2	APAAEVSW1	APAAEVSW0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Table 22.24 APAA0EVSW Register Contents

Bit Position	Bit Name	Reset Cause	Function
21 to 16	APAAEVSWMSK[5:0]	—	Mask writing to APAAEVSW[5:0]. Each bit is assigned to control each of APAAEVSW[5:0]. 0: Writing to APAAEVSW[n] is disabled. 1: Writing to APAAEVSW[n] is enabled. **Note:** These bits are read as 0.
5 to 0	APAAEVSW[5:0]	PRESETZ (after pclk*1)	The software event. Each bit corresponds to the software event ID (5 to 0). 0: Sets a software event (ID = bit) to low 1: Sets a software event (ID = bit) to high **Note:** The set value in this register becomes effective as an event input immediately after completion of writing. A software event is actually applied as an exception processing request in a pulse generation channel two cycles after completion of writing if the edge is detected (see **Figure 22.23**), and upon completion of writing if the level is detected (see **Figure 22.45**).

The following examples show how software events are configured.

- Writing 003F $003F_H$: All software events of channels 5 to 0 are set to 1 simultaneously.
- Writing 0000 $003F_H$: No effect (no software event)
- Writing 003F 0000_H: All software events of channels 5 to 0 are cleared to 0 simultaneously.
- Writing 0000 0000_H: No effect (The registers retain the previous values.)

22.4.4.3 APAA0EVSC — APAA0 Software Event Setting Register

This register sets the software event.

This register configures the operation options of the software event registers APAA0EVSW5 to 0.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	APAAEVSWCFG5[1:0]		APAAEVSWCFG4[1:0]		APAAEVSWCFG3[1:0]		APAAEVSWCFG2[1:0]		APAAEVSWCFG1[1:0]		APAAEVSWCFG0[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 22.25 APAA0EVSC Register Contents

Bit Position	Bit Name	Reset Cause	Function
11, 10	APAAEVSWCFG5[1:0]	PRESETZ	Configures the operation of software event register APAAEVSW5. 00: No control 01: One-shot (writing “high” to APAAEVSW5, and returning to “low” after one cycle of pclk) 10: One-shot (writing “low” to APAAEVSW5, and returning to “high” after one cycle of pclk) Others: Setting prohibited.
9, 8	APAAEVSWCFG4[1:0]	PRESETZ	Configures the operation of software event register APAAEVSW4. 00: No control 01: One-shot (writing “high” to APAAEVSW4, and returning to “low” after one cycle of pclk) 10: One-shot (writing “low” to APAAEVSW4, and returning to “high” after one cycle of pclk) Others: Setting prohibited.
7, 6	APAAEVSWCFG3[1:0]	PRESETZ	Configures the operation of software event register APAAEVSW3. 00: No control 01: One-shot (writing “high” to APAAEVSW3, and returning to “low” after one cycle of pclk) 10: One-shot (writing “low” to APAAEVSW3, and returning to “high” after one cycle of pclk) Others: Setting prohibited.
5, 4	APAAEVSWCFG2[1:0]	PRESETZ	Configures the operation of software event register APAAEVSW2. 00: No control 01: One-shot (writing “high” to APAAEVSW2, and returning to “low” after one cycle of pclk) 10: One-shot (writing “low” to APAAEVSW2, and returning to “high” after one cycle of pclk) Others: Setting prohibited.
3, 2	APAAEVSWCFG1[1:0]	PRESETZ	Configures the operation of software event register APAAEVSW1. 00: No control 01: One-shot (writing “high” to APAAEVSW1, and returning to “low” after one cycle of pclk) 10: One-shot (writing “low” to APAAEVSW1, and returning to “high” after one cycle of pclk) Others: Setting prohibited.
1, 0	APAAEVSWCFG0[1:0]	PRESETZ	Configures the operation of software event register APAAEVSW0. 00: No control 01: One-shot (writing “high” to APAAEVSW0, and returning to “low” after one cycle of pclk) 10: One-shot (writing “low” to APAAEVSW0, and returning to “high” after one cycle of pclk) Others: Setting prohibited.

CAUTION

When this register setting is modified, all pulse generation channels must be stopped ($APAA0CHEN = 0000_H$). If this register is modified during the operation of the pulse generation channels, event detection might operate in a wrong manner.

NOTES

When this register is modified to "01_B" or "10_B", the software event APAA0EVSW is set as follows.

1. When the setting of this register is changed to "01_B", the software event APAA0EVSW is set to a low level at the first PCLK after the setting change.
2. When the setting of this register is changed "10_B" the software event APAA0EVSW is set to a high level at the first PCLK after the setting change.

22.4.4.4 APAA0ESTA — APAA0 Event Status Register A

This register monitors the current level (high/low) of each bit of the event bus.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	APAAEBSTATLEV31	APAAEBSTATLEV30	APAAEBSTATLEV29	APAAEBSTATLEV28	APAAEBSTATLEV27	APAAEBSTATLEV26	APAAEBSTATLEV25	APAAEBSTATLEV24	APAAEBSTATLEV23	APAAEBSTATLEV22	APAAEBSTATLEV21	APAAEBSTATLEV20	APAAEBSTATLEV19	APAAEBSTATLEV18	APAAEBSTATLEV17	APAAEBSTATLEV16
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	APAAEBSTATLEV15	APAAEBSTATLEV14	APAAEBSTATLEV13	APAAEBSTATLEV12	APAAEBSTATLEV11	APAAEBSTATLEV10	APAAEBSTATLEV9	APAAEBSTATLEV8	APAAEBSTATLEV7	APAAEBSTATLEV6	APAAEBSTATLEV5	APAAEBSTATLEV4	APAAEBSTATLEV3	APAAEBSTATLEV2	APAAEBSTATLEV1	APAAEBSTATLEV0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 22.26 APAA0ESTA Register Contents

Bit Position	Bit Name	Reset Cause	Function
31 to 0	APAAEBSTATLEV [31:0]	PRESETZ	Indicate the current level of event bus eb_dat_r[31:0].

22.4.5 Control Register Details (Pulse Generation Channels 0 to 15)

These registers configure and read the operation of pulse generation channels 0 to 15.

Table 22.27 List of Control Registers (Pulse Generation Channels 0 to 15)

Offset Address	Symbol	Register Name	Value after Reset	Access Size	Function
0600_H	APAA0CCGA00	APAA0 Channel Setting Register A0	$0000\ 0100_H$	32	Setting of PGCH0
0604_H	APAA0CCGB00	APAA0 Channel Setting Register B0	$0000\ 0000_H$	32	Setting of PGCH0
0608_H	APAA0CSTA00	APAA0 Channel Status Register A0	$0000\ 0000_H$	32	Data of the current PGCH0 status
$060C_H$	APAA0CSTB00	APAA0 Channel Status Register B0	$0000\ 0000_H$	32	Element setting data currently effective for PGCH0
0610_H	APAA0CSTC00	APAA0 Channel Status Register C0	$0000\ 0000_H$	32	Element setting data currently effective for PGCH0
0614_H	APAA0CSTD00	APAA0 Channel Status Register D0	$0000\ 0000_H$	32	Element setting data currently effective for PGCH0
0618_H to $061C_H$	(N/A)		$0000\ 0000_H$	—	Unused area (read-only) **Note:** This area is read as 0 and cannot be modified.
0620_H	APAA0CCGA01	APAA0 Channel Setting Register A1	$0000\ 0100_H$	32	Setting of PGCH1
0624_H	APAA0CCGB01	APAA0 Channel Setting Register B1	$0000\ 0000_H$	32	Setting of PGCH1
0628_H	APAA0CSTA01	APAA0 Channel Status Register A1	$0000\ 0000_H$	32	Data of the current PGCH1 status
$062C_H$	APAA0CSTB01	APAA0 Channel Status Register B1	$0000\ 0000_H$	32	Element setting data currently effective for PGCH1
0630_H	APAA0CSTC01	APAA0 Channel Status Register C1	$0000\ 0000_H$	32	Element setting data currently effective for PGCH1
0634_H	APAA0CSTD01	APAA0 Channel Status Register D1	$0000\ 0000_H$	32	Element setting data currently effective for PGCH1
0638_H to $063C_H$	(N/A)		$0000\ 0000_H$	—	Unused area (read-only) **Note:** This area is read as 0 and cannot be modified.
(Details of PGCH2 to PGCH14 are the same as above, so they are omitted here.)					
$07E0_H$	APAA0CCGA15	APAA0 Channel Setting Register A15	$0000\ 0100_H$	32	Setting of PGCH15
$07E4_H$	APAA0CCGB15	APAA0 Channel Setting Register B15	$0000\ 0000_H$	32	Setting of PGCH15
$07E8_H$	APAA0CSTA15	APAA0 Channel Status Register A15	$0000\ 0000_H$	32	Data of the current PGCH15 status
$07EC_H$	APAA0CSTB15	APAA0 Channel Status Register B15	$0000\ 0000_H$	32	Element setting data currently effective for PGCH15
$07F0_H$	APAA0CSTC15	APAA0 Channel Status Register C15	$0000\ 0000_H$	32	Element setting data currently effective for PGCH15
$07F4_H$	APAA0CSTD15	APAA0 Channel Status Register D15	$0000\ 0000_H$	32	Element setting data currently effective for PGCH15
$07F8_H$ to $07FC_H$	(N/A)		$0000\ 0000_H$	—	Unused area (read-only) **Note:** This area is read as 0 and cannot be modified.

22.4.5.1 APAA0CCGAn — APAA0 Channel Setting Registers An (n = 00 to 15)

These registers are the setting registers for pulse generation channels n (0 to 15).

CAUTIONS

1. Only the APAACHNFSTEL[6:0] bits in this register (APAA0CCGAn) can be modified during operation. When modifying these bits during operation of a pulse generation channel, the previous value must be written to the other bits.
2. Modification of bits other than the APAACHNFSTEL[6:0] bits must be made while a pulse generation channel is stopped (APAA0CHEN.APAACMENPGCH[n] = 0). If a bit other than the APAACHNFSTEL[6:0] bits is modified during operation of a pulse generation channel, it may cause an unintended operation.
3. For the operation when the APAACHNFSTEL[6:0] bits are modified while a pulse generation channel is operating, see **Section 22.5.5.3, Changing Starting Element Number during Operation**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	APAACHNFSTEL[6:0]						
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	APAACHNCONT	APAACHNLEV	—	—	—	—	—	APAACHNINTSEL[2:0]		
Value after reset	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W

Table 22.28 APAA0CCGAn Register Contents

Bit Position	Bit Name	Reset Cause	Function
22 to 16	APAACHNFSTEL[6:0]	PRESETZ	Starting element number (00_H to $7F_H$)
9	APAACHNCONT	PRESETZ	Channel Continuation Setting 0: Single operation. The operation stops when the last element is completed. 1: Continuous operation. The operation restarts from the starting element number when the last element is completed.
8	APAACHNLEV	PRESETZ	The active level of an output pulse 0: Active-low 1: Active-high **Note:** This register setting is immediately reflected to the PWM output pin n (n: the corresponding bit).
2 to 0	APAACHNINTSEL[2:0]	PRESETZ	Interrupt setting. Interrupts of the following events are generated. An interrupt is a one-shot pulse (active-high). For the operation, see **(4) Interrupts** in **Section 22.5.4.2, Operation: Matching Comparator**. 000: After element transition 001: Upon channel operation completion 010: Upon ON match 011: Upon OFF match 100: Upon ON match or OFF match Others: Setting prohibited

Table 22.29 shows the correspondence between the register settings and the pulse generation channels.

Table 22.29 Correspondence between Pulse Generation Channels and Register Settings (APAA0CCGAn)

Pulse Generation Channel	Setting Register	Offset Address
PGCH0	APAA0CCGA00	0600_H
PGCH1	APAA0CCGA01	0620_H
PGCH2	APAA0CCGA02	0640_H
PGCH3	APAA0CCGA03	0660_H
PGCH4	APAA0CCGA04	0680_H
PGCH5	APAA0CCGA05	$06A0_H$
PGCH6	APAA0CCGA06	$06C0_H$
PGCH7	APAA0CCGA07	$06E0_H$
PGCH8	APAA0CCGA08	0700_H
PGCH9	APAA0CCGA09	0720_H
PGCH10	APAA0CCGA10	0740_H
PGCH11	APAA0CCGA11	0760_H
PGCH12	APAA0CCGA12	0780_H
PGCH13	APAA0CCGA13	$07A0_H$
PGCH14	APAA0CCGA14	$07C0_H$
PGCH15	APAA0CCGA15	$07E0_H$

22.4.5.2 APAA0CCGBn — APAA0 Channel Setting Registers Bn (n = 00 to 15)

These registers are setting registers for pulse generation channels n (0 to 15).

CAUTION

Modification of this register must be made while a pulse generation channel is stopped (APAA0CHEN.APAACMENPGCH[n] = 0).

Modifying this register during operation of a pulse generation channel may cause the channel to operate in an unintended manner.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	APAACHNEGRES[1:0]		APAACHNIDRES[4:0]				
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	APAACHNOPSTR	APAACHNEGSTR[1:0]		APAACHNIDSTR[4:0]					—	APAACHNEGSTP[1:0]		APAACHNIDSTP[4:0]				
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 22.30 APAA0CCGBn Register Contents (1/2)

Bit Position	Bit Name	Reset Cause	Function
22, 21	APAACHNEGRES[1:0]	PRESETZ	Indicate the effective edge(s) of the event input signal for the exception processing (restart). 00: Rising edge 01: Falling edge 10: Both (rising/falling) edges Others: Setting prohibited.
20 to 16	APAACHNIDRES[4:0]	PRESETZ	Specify the event bus signal to be used for the exception processing (restart). See **Table 22.31** for the correspondence between the values and event bus signals.
15	APAACHNOPSTR	PRESETZ	Option for the operation of the exception processing (start). This bit specifies the behavior of pulse generation channels after the operation is enabled (APAACMENPGCH[n] = 1). 0: Operation starts, triggered by a start exception after the operation is enabled. 1: Operation starts immediately after the operation is enabled. A start exception is not required. **Note:** When this option is set to 1, a stop exception does not stop the operation. (A stop exception stops the operation once, but the operation restarts from the initial element.) In this case, the operation can be stopped by setting the APPA0CHEN register to prohibit the operation.
14, 13	APAACHNEGSTR[1:0]	PRESETZ	Indicate the effective edge(s) of the event input signal for the exception processing (start). 00: Rising edge 01: Falling edge 10: Both (rising/falling) edges Others: Setting prohibited.
12 to 8	APAACHNIDSTR[4:0]	PRESETZ	Specify the event bus signal to be used for the exception processing (Start). See **Table 22.31** for the correspondence between the values and event bus signals.

Table 22.30 APAA0CCGBn Register Contents (2/2)

Bit Position	Bit Name	Reset Cause	Function
6, 5	APAACHNEGSTP[1:0]	PRESETZ	Indicate the effective edge(s) of the event input signal for the exception processing (stop). 00: Rising edge 01: Falling edge 10: Both (rising/falling) edges Others: Setting prohibited.
4 to 0	APAACHNIDSTP[4:0]	PRESETZ	Specify the event bus signal to be used for the exception processing (stop). See **Table 22.31** for the correspondence between the values and event bus signals.

The exception processing (start, stop, transfer, force, restart, and postpone) can use any internal event bus signals. The set value definitions are common to all types of exception processing.

Table 22.31 Correspondence between Exception Processing and Internal Event Bus Signals

Register Set Value	Selected Event	Related Setting (Register)	Register Set Value	Selected Event	Related Setting (Register)
00_H	PWM output (pgch_0)	None	10_H	External event input	APAA0EVSL00[5:0]*1
01_H	PWM output (pgch_1)	None	11_H	External event input	APAA0EVSL01[5:0]*1
02_H	PWM output (pgch_2)	None	12_H	External event input	APAA0EVSL02[5:0]*1
03_H	PWM output (pgch_3)	None	13_H	External event input	APAA0EVSL03[5:0]*1
04_H	PWM output (pgch_4)	None	14_H	External event input	APAA0EVSL04[5:0]*1
05_H	PWM output (pgch_5)	None	15_H	External event input	APAA0EVSL05[5:0]*1
06_H	PWM output (pgch_6)	None	16_H	External event input	APAA0EVSL06[5:0]]*1
07_H	PWM output (pgch_7)	None	17_H	External event input	APAA0EVSL07[5:0]]*1
08_H	PWM output (pgch_8)	None	18_H	External event input	APAA0EVSL08[5:0]]*1
09_H	PWM output (pgch_9)	None	19_H	External event input	APAA0EVSL09[5:0]]*1
$0A_H$	PWM output (pgch_10)	None	$1A_H$	External event input	APAA0EVSL10[5:0]]*1
$0B_H$	PWM output (pgch_11)	None	$1B_H$	External event input	APAA0EVSL11[5:0]]*1
$0C_H$	PWM output (pgch_12)	None	$1C_H$	External event input	APAA0EVSL12[5:0]]*1
$0D_H$	PWM output (pgch_13)	None	$1D_H$	External event input	APAA0EVSL13[5:0]]*1
$0E_H$	PWM output (pgch_14)	None	$1E_H$	External event input	APAA0EVSL14[5:0]]*1
$0F_H$	PWM output (pgch_15)	None	$1F_H$	External event input	APAA0EVSL15[5:0]]*1

Note 1. The external event sources 0 to 57 or the software events 0 to 5 which are specified by registers APAA0EVSL00 to APAA0EVSL15 are selected.

Table 22.32 Correspondence between Exception Processing and Setting Registers

Exception Processing	Setting Register	Remarks
Stop	APAACHNIDSTP[4:0]	Set to a pulse generation channel.
Start	APAACHNIDSTR[4:0]	Set to a pulse generation channel.
Restart	APAACHNIDRES[4:0]	Set to a pulse generation channel.
Transfer	APAAELMIDTRN[4:0]	Set to an element.
Force	APAAELMIDFOR[4:0]	Set to an element.
Postpone	APAAELMIDPPN[4:0]	Set to an element.

Note: For details on the exception processing, see **Section 22.5.1.3, Exception Processing**.

Table 22.33 shows the correspondence between the register settings and the pulse generation channels.

Table 22.33 Correspondence between Pulse Generation Channels and Register Settings (APAA0CCGBn)

Pulse Generation Channel	Setting Register	Offset Address
PGCH0	APAA0CCGB00	0604_H
PGCH1	APAA0CCGB01	0624_H
PGCH2	APAA0CCGB02	0644_H
PGCH3	APAA0CCGB03	0664_H
PGCH4	APAA0CCGB04	0684_H
PGCH5	APAA0CCGB05	$06A4_H$
PGCH6	APAA0CCGB06	$06C4_H$
PGCH7	APAA0CCGB07	$06E4_H$
PGCH8	APAA0CCGB08	0704_H
PGCH9	APAA0CCGB09	0724_H
PGCH10	APAA0CCGB10	0744_H
PGCH11	APAA0CCGB11	0764_H
PGCH12	APAA0CCGB12	0784_H
PGCH13	APAA0CCGB13	$07A4_H$
PGCH14	APAA0CCGB14	$07C4_H$
PGCH15	APAA0CCGB15	$07E4_H$

22.4.5.3 APAA0CSTAn — APAA0 Channel Status Registers An (n = 00 to 15)

These registers monitor the channel status data for pulse generation channels n (0 to 15).

CAUTION

All the effective bits in these registers except APAACHNEN and APAAEOPWM take effect only when the reference and event buses are enabled (APAACMENAPA = 1) and the operation of the target channel is enabled (APAA0EN[n] = 1) (this setting is hereinafter described as “channel-enabled setting”).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	APAACHNCUREL[6:0]						
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	APAACMPSTAT[4:0]					—	—	APAAMATSTAT[1:0]		APAATRNSTAT	APAAPPNSTAT	APAACHNEN	APAAEOPWM
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 22.34 APAA0CSTAn Register Contents (1/2)

Bit Position	Bit Name	Reset Cause	Function
22 to 16	APAACHNCUREL[6:0]	PRESETZ	Current element number (the element number currently effective for the pulse generation channel) **Note:** The value is retained except when the channel-enabled setting is made.
12 to 8	APAACMPSTAT[4:0]	PRESETZ	Indicate the current operation status of the matching comparator. See **Figure 22.52** and **Figure 22.13**. 00000: disable 00001: dis_wstart 00010: Unused 00011: pre_dis 00100: pre_on_cur 00101: pre_on_init 00110: pre_on_nxt 00111: pre_off 01000: waiton 01001: waiton_tron 01010: forcedoff_troff 01011: on_postponed 01100: waitoff 01101: waitoff_tron 01110: matoff_troff 01111: off_postponed 10000: terminate **Note:** The value is 00000 (disable) except when the channel-enabled setting is made.
5, 4	APAAMATSTAT[1:0]	PRESETZ	Indicate the matching detection status in the matching comparator. 00: Not match (matching is not detected) 01: ON match is detected. 10: OFF match is detected. 11: (Nothing detected) **Note:** The value is 00 (not match) except when the channel-enabled setting is made.

Table 22.34 APAA0CSTAn Register Contents (2/2)

Bit Position	Bit Name	Reset Cause	Function
3	APAATRNSTAT	PRESETZ	Indicates whether the transfer exception is entered (whether element transition upon OFF match is enabled) in the currently effective element. 0: Transfer exception is not entered. 1: Transfer exception is entered. **Note:** The value is 0 (transfer exception not entered) except when the channel-enabled setting is made.
2	APAAPPNSTAT	PRESETZ	Indicates whether the currently effective element is in the Postpone state. 0: Postpone OFF 1: Postpone ON **Note:** The value is 0 (Postpone OFF) except when the channel-enabled setting is made.
1	APAACHNEN	PRESETZ	The operation status of the corresponding pulse generation channel. 0: Disabled: PGCHn (n = 0 to 15) is stopped. 1: Enabled: PGCHn (n = 0 to 15) is in operation. **Note:** The bit contents are identical to those of bits APAACMENPGCH[n] in the APAA0CHEN register, so the same data can be read. The above feature is effective even if the channel-enabled setting is not made.
0	APAAEOPWM	PRESETZ	The operation status of the corresponding pulse generation channel (PWM). This bit indicates the output level of the PWM output pin n of the corresponding pulse generation channel. **Note:** The above feature is effective even if the channel-enabled setting is not made.

Table 22.35 shows the correspondence between the register settings and the pulse generation channels.

Table 22.35 Correspondence between Pulse Generation Channels and Register Settings (APAA0CSTAn)

Pulse Generation Channel	Setting Register	Offset Address
PGCH0	APAA0CSTA00	0608_H
PGCH1	APAA0CSTA01	0628_H
PGCH2	APAA0CSTA02	0648_H
PGCH3	APAA0CSTA03	0668_H
PGCH4	APAA0CSTA04	0688_H
PGCH5	APAA0CSTA05	$06A8_H$
PGCH6	APAA0CSTA06	$06C8_H$
PGCH7	APAA0CSTA07	$06E8_H$
PGCH8	APAA0CSTA08	0708_H
PGCH9	APAA0CSTA09	0728_H
PGCH10	APAA0CSTA10	0748_H
PGCH11	APAA0CSTA11	0768_H
PGCH12	APAA0CSTA12	0788_H
PGCH13	APAA0CSTA13	$07A8_H$
PGCH14	APAA0CSTA14	$07C8_H$
PGCH15	APAA0CSTA15	$07E8_H$

22.4.5.4 APAA0CSTBn — APAA0 Channel Status Registers Bn (n = 00 to 15)

These registers monitor the element settings for pulse generation channels n (0 to 15).

The values that can be monitored by these registers are element information that is currently effective in the corresponding pulse generation channel, and those values are loaded from the element RAM.

CAUTION

All the effective bits in these registers take effect only when the reference and event buses are enabled (APAACMENAPA = 1) and the operation of the target channel is enabled (APAA0EN[n] = 1) (this setting is hereinafter described as “channel-enabled setting”).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	APAAELDFINEL	APAAELDCONT	—	—	—	—	—	APAAELDRBIDON[3:0]				APAAELDMCON[1:0]		APAAELDABREON
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	APAAELDRBDATON[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 22.36 APAA0CSTBn Register Contents (1/2)

Bit Position	Bit Name	Reset Cause	Function
29	APAAELDFINEL	PRESETZ	The last element flag. This flag indicates whether the element is the last one. 0: The element is not the last one. 1: The element is the last one. **Note:** The value is retained except when the channel-enabled setting is made.
28	APAAELDCONT	PRESETZ	The element continuation setting. This bit indicates how the currently enabled element operates after an OFF match. 0: Single operation (matching comparison is stopped until transition to the next element.) 1: Continuous operation (the operation returns to ON matching comparison after an OFF match is detected.) **Note:** The value is retained except when the channel-enabled setting is made.
22 to 19	APAAELDRBIDON[3:0]	PRESETZ	Indicate the reference number for ON matching comparison. For the correspondence between the values and the reference numbers, see **Table 22.37**. **Note:** The value is retained except when the channel-enabled setting is made.
18, 17	APAAELDMCON[1:0]	PRESETZ	Indicate the criteria for ON matching comparison. 00: [The specified reference value] == [Comparison baseline APAAELDRBDATON[15:0]] 01: [The specified reference value] > [Comparison baseline APAAELDRBDATON[15:0]] 10: [The specified reference value] < [Comparison baseline APAAELDRBDATON[15:0]] Others: Setting prohibited. **Note:** The value is retained except when the channel-enabled setting is made. The settings of 01 (>) and 10 (<) cannot be used when an overflow is generated at the reference value (the values before and after the overflow cannot be compared properly).

Table 22.36 APAA0CSTBn Register Contents (2/2)

Bit Position	Bit Name	Reset Cause	Function
16	APAAELDABREON	PRESETZ	Indicates whether the specified baseline (APAAELDRBDATON) is interpreted as an absolute value or a relative value for determining the comparison baseline (APAAELDRBDATON) for ON matching comparison. 0: Absolute designation. The specified baseline is interpreted as an absolute value. 1: Relative designation. The baseline is interpreted as the difference from the reference value at the time of the start of matching comparison. However, the relative designation is available only for references 0, 1, and sw. **Note:** The value is retained except when the channel-enabled setting is made.
15 to 0	APAAELDRBDATON[15:0]	PRESETZ	Indicate the comparison baseline for ON match. A comparison baseline is used for actual matching comparison, and has been converted to an absolute value. The value differs depending on the APAAELDRBDATON setting. • APAAELDABREON = 0 (absolute designation) APAAELDRBDATON[15:0] = APAAELMRBDATON[15:0] • APAAELDABREON = 1 (relative designation) APAAELDRBDATON[15:0] = refb_dat_n[15:0] + APAAELMRBDATON[15:0] **Note:** If OFF matching comparison is in progress when these bits are read, or the relative designation is made for the references 2 to 9, the values of these bits are identical to the relatively designated values. APAAELDRBDATON[15:0] = APAAELMRBDATON[15:0] The value is retained except when the channel-enabled setting is made.

For both ON match and OFF match, the correspondence between the compared reference numbers and setting values is as listed in **Table 22.37**. The relative destination is only available for references 0, 1, and sw.

Table 22.37 Comparison between Matching Comparison Setting and Reference Numbers

eln_rbid_on/off Set Value	Reference No.	Reference Name	Time Resolution	Relative Designation
0_H	0	Reference 0	pclk*4	Possible
1_H	1	Reference 1	pclk*4	Possible
2_H	2	Reference 2	pclk*20	Not possible*1
3_H	3	Reference 3	pclk*20	Not possible*1
4_H	4	Reference 4	pclk*20	Not possible*1
5_H	5	Reference 5	pclk*20	Not possible*1
6_H	6	Reference 6	pclk*20	Not possible*1
7_H	7	Reference 7	pclk*20	Not possible*1
8_H	8	Reference 8	pclk*20	Not possible*1
9_H	9	Reference 9	pclk*20	Not possible*1
F_H	f	Software reference	pclk*20	Possible
Others	0	(Interpreted as reference 0)	pclk*4	Possible

Note 1. If the relative designation is set for a reference for which the relative designation is not possible, the setting is ignored (the comparison baseline is the same as that for the absolute designation).

Table 22.38 shows the correspondence between the register settings and pulse generation channels.

Table 22.38 Correspondence between Pulse Generation Channels and Register Settings (APAA0CSTBn)

Pulse Generation Channel	Setting Register	Offset Address
PGCH0	APAA0CSTA00	$060C_H$
PGCH1	APAA0CSTA01	$062C_H$
PGCH2	APAA0CSTA02	$064C_H$
PGCH3	APAA0CSTA03	$066C_H$
PGCH4	APAA0CSTA04	$068C_H$
PGCH5	APAA0CSTA05	$06AC_H$
PGCH6	APAA0CSTA06	$06CC_H$
PGCH7	APAA0CSTA07	$06EC_H$
PGCH8	APAA0CSTA08	$070C_H$
PGCH9	APAA0CSTA09	$072C_H$
PGCH10	APAA0CSTA10	$074C_H$
PGCH11	APAA0CSTA11	$076C_H$
PGCH12	APAA0CSTA12	$078C_H$
PGCH13	APAA0CSTA13	$07AC_H$
PGCH14	APAA0CSTA14	$07CC_H$
PGCH15	APAA0CSTA15	$07EC_H$

22.4.5.5 APAA0CSTCn — APAA0 Channel Status Registers Cn (n = 00 to 15)

These registers monitor the element settings for the pulse generation channels n (0 to 15).

The values that can be monitored by these registers are element information that is currently effective in the corresponding pulse generation channel, and those values are loaded from the element RAM.

CAUTION

All the effective bits in these registers take effect only when the reference and event buses are enabled (APAACMENAPA = 1) and the operation of the target channel is enabled (APAA0EN[n] = 1) (this setting is hereinafter described as “channel-enabled setting”).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	APAAELDRBIDOFF[3:0]				APAAELD_MCOFF[1:0]		APAAELDABREOFF
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	APAAELDRBDATOFF[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 22.39 APAA0CSTCn Register Contents (1/2)

Bit Position	Bit Name	Reset Cause	Function
22 to 19	APAAELDRBIDOFF[3:0]	PRESETZ	Indicate the reference number for OFF matching comparison. For the correspondence between the values and the reference numbers, see **Table 22.37**. **Note:** The value is retained except when the channel-enabled setting is made.
18, 17	APAAELD_MCOFF[1:0]	PRESETZ	Indicate the condition for OFF matching comparison. 00: [The specified reference value] == [Comparison baseline APAAELDRBDATON[15:0]] 01: [The specified reference value] > [Comparison baseline APAAELDRBDATON[15:0]] 10: [The specified reference value] < [Comparison baseline APAAELDRBDATON[15:0]] Others: Setting prohibited. **Note:** The value is retained except when the channel-enabled setting is made. The settings of 01 (>) and 10 (<) cannot be used when an overflow is generated at the reference value. (The values before and after the overflow cannot be compared properly.)
16	APAAELDABREOFF	PRESETZ	Indicates whether the specified baseline (APAAELDRBDATOFF) is interpreted as an absolute value or a relative value for determining the comparison baseline (APAAELDRBDATOFF) for OFF matching comparison. 0: Absolute designation. The specified baseline is interpreted as an absolute value. 1: Relative designation. The baseline is interpreted as the difference from the reference value at the time of the start of matching comparison. However, the relative designation is available only for references 0, 1, and sw. **Note:** The value is retained except when the channel-enabled setting is made.

Table 22.39 APAA0CSTCn Register Contents (2/2)

Bit Position	Bit Name	Reset Cause	Function
15 to 0	APAAELDRBDATOFF[15:0]	PRESETZ	Indicate the comparison baseline for OFF match. A comparison baseline is used for actual matching comparison, and has been converted to an absolute value. The value differs depending on the APAAELDABREOFF setting. • APAAELDABREOFF = 0 (absolute designation) APAAELDRBDATOFF[15:0] = APAAELMRBDATOFF[15:0] • APAAELDABREOFF = 1 (relative designation) APAAELDRBDATOFF[15:0] = refb_dat_n[15:0] + APAAELMRBDATOFF[15:0] **Note:** If ON matching comparison is in progress when these bits are read, or the relative designation is made for the references 2 to 9, the values of these bits are not converted to absolutely designated values and are identical to the relatively designated values. APAAELDRBDATOFF[15:0] = APAAELMRBDATOFF[15:0] The value is retained except when the channel-enabled setting is made.

Table 22.40 shows the correspondence between the register settings and pulse generation channels.

Table 22.40 Correspondence between Pulse Generation Channels and Register Settings

Pulse Generation Channel	Setting Register	Offset Address
PGCH0	APAA0CSTC00	0610_H
PGCH1	APAA0CSTC01	0630_H
PGCH2	APAA0CSTC02	0650_H
PGCH3	APAA0CSTC03	0670_H
PGCH4	APAA0CSTC04	0690_H
PGCH5	APAA0CSTC05	$06B0_H$
PGCH6	APAA0CSTC06	$06D0_H$
PGCH7	APAA0CSTC07	$06F0_H$
PGCH8	APAA0CSTC08	0710_H
PGCH9	APAA0CSTC09	0730_H
PGCH10	APAA0CSTC10	0750_H
PGCH11	APAA0CSTC11	0770_H
PGCH12	APAA0CSTC12	0790_H
PGCH13	APAA0CSTC13	$07B0_H$
PGCH14	APAA0CSTC14	$07D0_H$
PGCH15	APAA0CSTC15	$07F0_H$

22.4.5.6 APAA0CSTDn — APAA0 Channel Status Registers Dn (n = 00 to 15)

These registers monitor the element settings for the pulse generation channels n (0 to 15).

The values that can be monitored by these registers are the element information that is currently effective in the corresponding pulse generation channel, and those values are loaded from the element RAM.

CAUTION

All the effective bits in these registers take effect only when the reference and event buses are enabled (APAACMENAPA = 1) and the operation of the target channel is enabled (APAA0EN[n] = 1) (this setting is hereinafter described as “channel-enabled setting”).

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	APAAELDNEXTEL[6:0]							APAAELDOPTRN	APAAELDEGTRN[1:0]		APAAELDIDTRN[4:0]				
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	APAAELDOPFOR	APAAELDEGFOR[1:0]		APAAELDIDFOR[4:0]					—	—	APAAELDLVPPN	APAAELDIDPPN[4:0]				
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 22.41 APAA0CSTDn Register Contents (1/2)

Bit Position	Bit Name	Reset Cause	Function
30 to 24	APAAELDNEXTEL[6:0]	PRESETZ	The next element number (00_H to $7F_H$). **Note:** The value is retained except when the channel-enabled setting is made.
23	APAAELDOPTRN	PRESETZ	Option for the operation of the exception processing (Transfer). Indicates whether an element transition is allowed without a Transfer exception in a certain condition. 0: The input of the Transfer exception enables the element transition to occur upon an OFF match. Once the Transfer exception is input, the element transition occurs only for the currently effective elements upon an OFF match. When the element is updated, another Transfer exception must be input. 1: The occurrence of an OFF match automatically triggers transition to next elements. Enabling of the element transition using the Transfer exception is not required. The element transition is given priority over the element continuation setting (APAAELMCONT). **Note:** The value is retained except when the channel-enabled setting is made.
22, 21	APAAELDEGTRN[1:0]	PRESETZ	Indicate the effective edge(s) of the event input signal for the exception processing (Transfer). 00: Rising edge 01: Falling edge 10: Both (rising/falling) edges Others: Setting prohibited. **Note:** The value is retained except when the channel-enabled setting is made.

Table 22.41 APAA0CSTDn Register Contents (2/2)

Bit Position	Bit Name	Reset Cause	Function
20 to 16	APAAELDIDTRN[4:0]	PRESETZ	Specify the event bus signal to be used for the exception processing (transfer). See **Table 22.31** for the correspondence between the values and event bus signals. **Note:** The value is retained except when the channel-enabled setting is made.
15	APAAELDOPFOR	PRESETZ	Option for the operation of the exception processing (force). 0: Next match is forcibly generated. ON waiting → ON match generated + transition to OFF waiting OFF waiting → OFF match forcibly generated + transition to the next state 1: OFF match is forcibly generated. ON waiting → OFF match generated. OFF waiting → ON match generated. **Note:** The value is retained except when the channel-enabled setting is made.
14, 13	APAAELDEGFOR[1:0]	PRESETZ	Indicate the effective edge(s) of the event input signal for the exception processing (force). 00: Rising edge 01: Falling edge 10: Both (rising/falling) edges Others: Setting prohibited. **Note:** The value is retained except when the channel-enabled setting is made.
12 to 8	APAAELDIDFOR[4:0]	PRESETZ	Specify the event bus signal to be used for the exception processing (force). See **Table 22.31** for the correspondence between the values and event bus signals. **Note:** The value is retained except when the channel-enabled setting is made.
5	APAAELDLVPPN	PRESETZ	Indicates the effective level of the event input signal for the exception processing (postpone). 0: Low level 1: High level **Note:** The value is retained except when the channel-enabled setting is made.
4 to 0	APAAELDIDPPN[4:0]	PRESETZ	Specify the event bus signal to be used for the exception processing (postpone). See **Table 22.31** for the correspondence between the values and event bus signals. **Note:** The value is retained except when the channel-enabled setting is made.

Table 22.42 shows the correspondence between the register settings and pulse generation channels.

Table 22.42 Correspondence between Pulse Generation Channels and Register Settings (APAA0CSTDn)

Pulse Generation Channel	Register	Offset Address
PGCH0	APAA0CSTD00	0614_H
PGCH1	APAA0CSTD01	0634_H
PGCH2	APAA0CSTD02	0654_H
PGCH3	APAA0CSTD03	0674_H
PGCH4	APAA0CSTD04	0694_H
PGCH5	APAA0CSTD05	$06B4_H$
PGCH6	APAA0CSTD06	$06D4_H$
PGCH7	APAA0CSTD07	$06F4_H$
PGCH8	APAA0CSTD08	0714_H
PGCH9	APAA0CSTD09	0734_H
PGCH10	APAA0CSTD10	0754_H
PGCH11	APAA0CSTD11	0774_H
PGCH12	APAA0CSTD12	0794_H
PGCH13	APAA0CSTD13	$07B4_H$
PGCH14	APAA0CSTD14	$07D4_H$
PGCH15	APAA0CSTD15	$07F4_H$

22.4.6 Details of Elements

These registers configure and read the elements.

There are 128 elements (00_H to $7F_H$) available, and they can be read and written through the P-Bus.

The elements are deployed on the SRAM, so that they cannot be initialized by a reset.

Table 22.43 List of Element Setting Registers

Offset Address	Symbol	Register Name	Value after Reset	Access Size	Function
1000_H	APAA0ELMA000	APAA0 Element Setting Register A0		32	Setting data 0 of element 00
1004_H	APAA0ELMB000	APAA0 Element Setting Register B0		32	Setting data 1 of element 00
1008_H	APAA0ELMC000	APAA0 Element Setting Register C0		32	Setting data 2 of element 00
$100C_H$	(N/A)		$0000\ 0000_H$		Unused area (read-only) **Note:** This area is read as 0 and cannot be modified.
1010_H	APAA0ELMA001	APAA0 Element Setting Register A1		32	Setting data 0 of element 01
1014_H	APAA0ELMB001	APAA0 Element Setting Register B1		32	Setting data 1 of element 01
1018_H	APAA0ELMC001	APAA0 Element Setting Register C1		32	Setting data 2 of element 01
$101C_H$	(N/A)		$0000\ 0000_H$		Unused area (read-only) **Note:** This area is read as 0 and cannot be modified.
(Details of 1020_H to $17DC_H$ are the same as above, so they are omitted here.)					
$17E0_H$	APAA0ELMA126	APAA0 Element Setting Register A126		32	Setting data 0 of element 7E
$17E4_H$	APAA0ELMB126	APAA0 Element Setting Register B126		32	Setting data 1 of element 7E
$17E8_H$	APAA0ELMC126	APAA0 Element Setting Register C126		32	Setting data 2 of element 7E
$17EC_H$	(N/A)		$0000\ 0000_H$		Unused area (read-only) **Note:** This area is read as 0 and cannot be modified.
$17F0_H$	APAA0ELMA127	APAA0 Element Setting Register A127		32	Setting data 0 of element 7F
$17F4_H$	APAA0ELMB127	APAA0 Element Setting Register B127		32	Setting data 1 of element 7F
$17F8_H$	APAA0ELMC127	APAA0 Element Setting Register C127		32	Setting data 2 of element 7F
$17FC_H$	(N/A)		$0000\ 0000_H$		Unused area (read-only) **Note:** This area is read as 0 and cannot be modified.
1800_H to $1FFC_H$	(N/A)		$0000\ 0000_H$		Unused area **Note:** When accessed, the area from 1000_H to $17FC_H$ is read from or written to.

22.4.6.1 APAA0ELMAn — APAA0 Element Setting Registers An (n = 000 to 127)

These registers configure the operation of elements.

CAUTIONS

1. Since the contents of these registers are mapped to the SRAM, they are not initialized by hardware (values after reset are undefined). Therefore, the SRAM must be initialized before the APA starts operating even if an element is not currently used.
2. For writing to and reading from these registers, see **Section 22.5.6.2, Write to Element RAM**, and **Section 22.5.6.3, Read from Element RAM**, respectively.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	APAAELMFINEL	APAAELMCONT	—	—	—	—	—	APAAELMRBIDON[3:0]				APAAELMMCON[1:0]		APAAELMABREON
Value after reset	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
R/W	R	R	R/W	R/W	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	APAAELMRBDATON[15:0]															
Value after reset	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 22.44 APAA0ELMAn Register Contents (1/2)

Bit Position	Bit Name	Reset Cause	Function
29	APAAELMFINEL	(None)	Last element setting. This bit specifies whether the element is to be the last one. 0: The element is not the last one. 1: The element is the last one.
28	APAAELMCONT	(None)	Element continuation setting. This bit specifies how the currently-effective element operates after an OFF match. 0: Single operation (matching comparison is stopped until transition to next elements.) 1: Continuous operation (the operation returns to ON matching comparison after an OFF match is detected.)
22 to 19	APAAELMRBIDON[3:0]	(None)	Specify the reference number for ON matching comparison. For the correspondence between the values and the reference numbers, see **Table 22.37**.
18, 17	APAAELMMCON[1:0]	(None)	Specify the condition for ON matching comparison. Comparison of large and small values is performed by interpreting the reference value as an unsigned number. The specified baseline is converted to an absolute designation format before comparison (see the APAAELDRBDATON[15:0] bits). 00: [Specified reference value] == [Comparison baseline APAAELDRBDATON[15:0]] 01: [Specified reference value] > [Comparison baseline APAAELDRBDATON[15:0]] 10: [Specified reference value] < [Comparison baseline APAAELDRBDATON[15:0]] Others: Setting prohibited

Table 22.44 APAA0ELMAn Register Contents (2/2)

Bit Position	Bit Name	Reset Cause	Function
16	APAAELMABREON	(None)	Indicates whether the specified baseline (APAAELMRBDATON) is interpreted as an absolute value or a relative value for determining the comparison baseline (APAAELMRBDATON) for ON matching comparison. 0: Absolute designation. The specified baseline is interpreted as an absolute value. 1: Relative designation. The baseline is interpreted to specify the difference from the reference value at the time when matching comparison started. Relative designation is valid only for references 0, 1, and sw.
15 to 0	APAAELMRBDATON[15:0]	(None)	Specify the baseline for ON matching comparison. It is specified as an unsigned value.

22.4.6.2 APAA0ELMBn — APAA0 Element Setting Registers Bn (n = 000 to 127)

These registers configure the operation of elements.

CAUTIONS

1. Since the contents of these registers are mapped to the SRAM, they are not initialized by hardware (values after reset are undefined). Therefore, the SRAM must be initialized before the APA starts operating even if an element is not currently used.
2. For writing to and reading from theses registers, see **Section 22.5.6.2, Write to Element RAM**, and **Section 22.5.6.3, Read from Element RAM**, respectively.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	APAAELMRBIDOFF[3:0]				APAAELMMCOFF[1:0]		APAAELMABREOFF
Value after reset	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	APAAELMRBDATOFF[15:0]															
Value after reset	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 22.45 APAA0ELMBn Register Contents

Bit Position	Bit Name	Reset Cause	Function
22 to 19	APAAELMRBIDOFF[3:0]	(None)	Specify the reference number for OFF matching comparison. For the correspondence between the values and the reference numbers, see **Table 22.37**.
18, 17	APAAELMMCOFF[1:0]	(None)	Specify the condition for OFF matching comparison. Comparison of large and small values is performed by interpreting the reference value as an unsigned number. The specified baseline is converted to an absolute designation format before comparison (see the APAAELDRBDATOFF[15:0] bits). 00: [Specified reference value] == [Comparison baseline APAAELDRBDATON[15:0]] 01: [Specified reference value] > [Comparison baseline APAAELDRBDATON[15:0]] 10: [Specified reference value] < [Comparison baseline APAAELDRBDATON[15:0]] Others: Setting prohibited
16	APAAELMABREOFF	(None)	Indicates whether the specified baseline (APAAELMRBDATOFF) is interpreted as an absolute value or a relative value for determining the comparison baseline (APAAELMRBDATOFF) for OFF matching comparison. 0: Absolute designation. The specified baseline is interpreted as an absolute value. 1: Relative designation. The baseline is interpreted to specify the difference from the reference value at the time when matching comparison started. Relative designation is valid only for references 0, 1, and sw.
15 to 0	APAAELMRBDATOFF[15:0]	(None)	Specify the baseline for OFF matching comparison. It is specified as an unsigned value.

22.4.6.3 APAA0ELMCn — APAA0 Element Setting Registers Cn (n = 000 to 127)

These registers configure the operation of elements.

CAUTIONS

1. Since the contents of these registers are mapped to the SRAM, they are not initialized by hardware (values after reset are undefined). Therefore, the SRAM must be initialized before the APA starts operating even if an element is not currently used.
2. For writing to and reading from these registers, see **Section 22.5.6.2, Write to Element RAM**, and **Section 22.5.6.3, Read from Element RAM**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	APAAELMNEXTEL[6:0]							APAAELMOPTRN	APAAELMEGTRN		APAAELMIDTRN[4:0]				
Value after reset	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	APAAELMOPFOR	APAAELMEGFOR		APAAELMIDFOR[4:0]					—	—	APAAELMLVPPN	APAAELMIDPPN[4:0]				
Value after reset	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined	Undefined
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 22.46 APAA0ELMCn Register Contents (1/2)

Bit Position	Bit Name	Reset Cause	Function
30 to 24	APAAELMNEXTEL[6:0]	(None)	Specify the next element number (00_H to $7F_H$).
23	APAAELMOPTRN	(None)	Option for the operation of the exception processing (transfer). Specifies whether an element transition is allowed without a transfer exception in a certain condition. 0: the input of the transfer exception enables the element transition to occur upon an OFF match. Once the transfer exception is input, the element transition occurs only for the currently effective elements upon an OFF match. After the element update, another transfer exception must be input. 1: the occurrence of an OFF match automatically triggers transition to next elements, only when element continuation is disabled (APAAELMCONT = 0). Enabling of the element transition by the transfer exception is not required.
22, 21	APAAELMEGTRN	(None)	Indicate the effective edge(s) of the event input signal for the exception processing (transfer). 00: Rising edge 01: Falling edge 10: Both (rising/falling) edges Others: Setting prohibited.
20 to 16	APAAELMIDTRN[4:0]	(None)	Specify the event bus signal to be used for the exception processing (transfer). See **Table 22.31** for the correspondence between the values and event bus signals.

Table 22.46 APAA0ELMCn Register Contents (2/2)

Bit Position	Bit Name	Reset Cause	Function
15	APAAELMOPFOR	(None)	Option for the operation of for the exception processing (force). 0: Next match is forcibly generated. ON waiting → ON match generated + transition to OFF waiting OFF waiting → OFF match forcibly generated + transition to the next state 1: OFF match is forcibly generated. ON waiting → OFF match generated OFF waiting → ON match generated
14, 13	APAAELMEGFOR	(None)	Indicate the effective edge(s) of the event input signal for the exception processing (Force). 00: Rising edge 01: Falling edge 10: Both (rising/falling) edges Others: Setting prohibited.
12 to 8	APAAELMIDFOR[4:0]	(None)	Specify the event bus signal to be used for the exception processing (force). See **Table 22.31** for the correspondence between the values and event bus signals.
5	APAAELMLVPPN	(None)	Indicates the effective level of the event input assigned to the exception processing (postpone). 0: Low level 1: High level
4 to 0	APAAELMIDPPN[4:0]	(None)	Specify the event bus signal to be used for the exception processing (postpone). See **Table 22.31** for the correspondence between the values and event bus signals.

22.5 Functions

22.5.1 Concept

22.5.1.1 Pulse Generation by APA

The APA generates pulses in the following manner:

- The APA generates the pulse waveforms by continuously comparing the external inputs (references) against a pair of specified criteria: the criterion for making the PWM output ON (ON matching condition) and the criterion for making the PWM output OFF (OFF matching condition). The state in which ON match is awaited and the state in which OFF match is awaited are mutually exclusive.
- There are the following two ways to specify the matching conditions for ON match and OFF match.
  - Absolute designation: The matching condition is specified by a value against which the reference values are directly compared.
  - Relative designation: The matching condition is specified by a relative value variation from the reference value at the time when the matching comparison starts.
- Both cyclic pulses and one-shot pulses can be generated by specifying the matching conditions.
- The pulse generation conditions are specified by data structures called elements. More information on elements will be given below.
- For each APA module, there are 16 channels of the comparison circuit (pulse generation channel), each of which can operate independently.

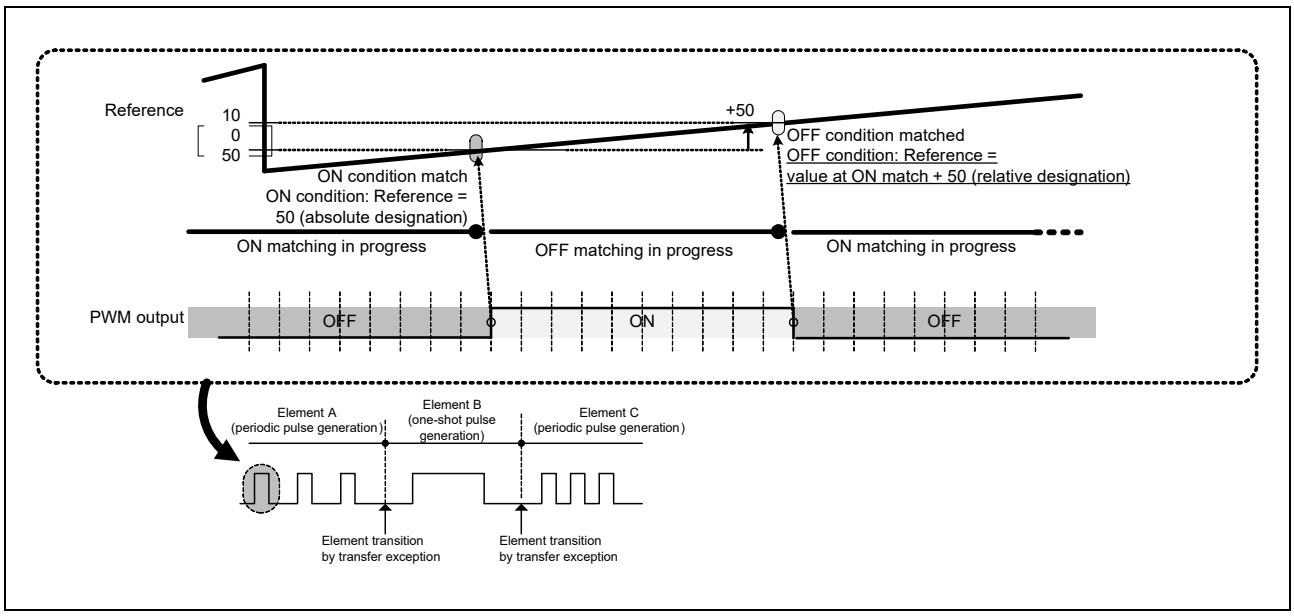


Figure 22.5 Conceptual Diagram of PWM Generation by APA

22.5.1.2 Element

- A data structure of a set of parameters that specify the pulse generation conditions is called an element. Each pulse generation channel loads a specific element, and generates a PWM waveform according to the element contents.
- For data specified for each element, see **Section 22.4.6.1, APAA0ELMAn — APAA0 Element Setting Registers An (n = 000 to 127)**, **Section 22.4.6.2, APAA0ELMBn — APAA0 Element Setting Registers Bn (n = 000 to 127)**, and **Section 22.4.6.3, APAA0ELMCn — APAA0 Element Setting Registers Cn (n = 000 to 127)**.
- A total of 128 elements can be specified, and they can be shared among different pulse generation channels (PGCH0 to PGCH15).
- An element has a pointer to the next elements, and transition to next elements can be controlled by setting conditions (see **Section 22.5.1.3, Exception Processing**). By this mechanism, a single pulse generation channel can use multiple elements, so that a pulse generation channel can generate a complex pulse sequence by combination of elements.

22.5.1.3 Exception Processing

- When a pulse generation channel generates pulses by PWM, the internal operation is controlled by a state machine. A signal that triggers a specific state transition is called exception, and the transition itself is called the exception processing.
- Any event input can be assigned as an exception, and an edge (rising/falling) or a level (high/low) can be assigned for each type of exception processing. Some exception processing types (stop, start, and restart) are defined for each pulse generation channel, and others (transfer, force, and postpone) are defined for each element.
- There are six exceptions processing as listed in **Table 22.47**.

Table 22.47 Exception Processing

Exception Processing	Operation	Assignment to Event Input	Defined for	Remarks
Stop	Stops the matching comparison.	Edge (rising /falling/both)	Pulse generation channel	
Start	Starts the matching comparison.	Edge (rising /falling/both)	Pulse generation channel	ON matching comparison starts based on the setting in the initial element defined for each pulse generation channel.
Restart	Initiate a transition to the initial element.	Edge (rising /falling/both)	Pulse generation channel	ON matching comparison starts based on the setting in the initial element defined for each pulse generation channel.
Transfer	Permits an element transition upon OFF match.	Edge (rising /falling/both)	Element	After a transfer exception input, the permission of element transition upon OFF match is retained. This condition is valid until an element transition occurs (see the option bit APAAELMOPTRN).
Force	Forcibly generates ON match or OFF match.	Edge (rising /falling/both)	Element	By default, the next ON/OFF match is forcibly generated (see the option bit APAAELMOPFOR).
Postpone	Temporarily postpones the operation that normally takes place upon match detection.	Level (high/low)	Element	Postpones the reflection in the PWM output pin, switching ON/OFF conditions and its accompanying element transition, which should normally occur upon a match. Matching comparison itself is in progress, and if there is a match during the Postpone period, a given operation restarts after release from the Postpone state.

22.5.2 Reference Control

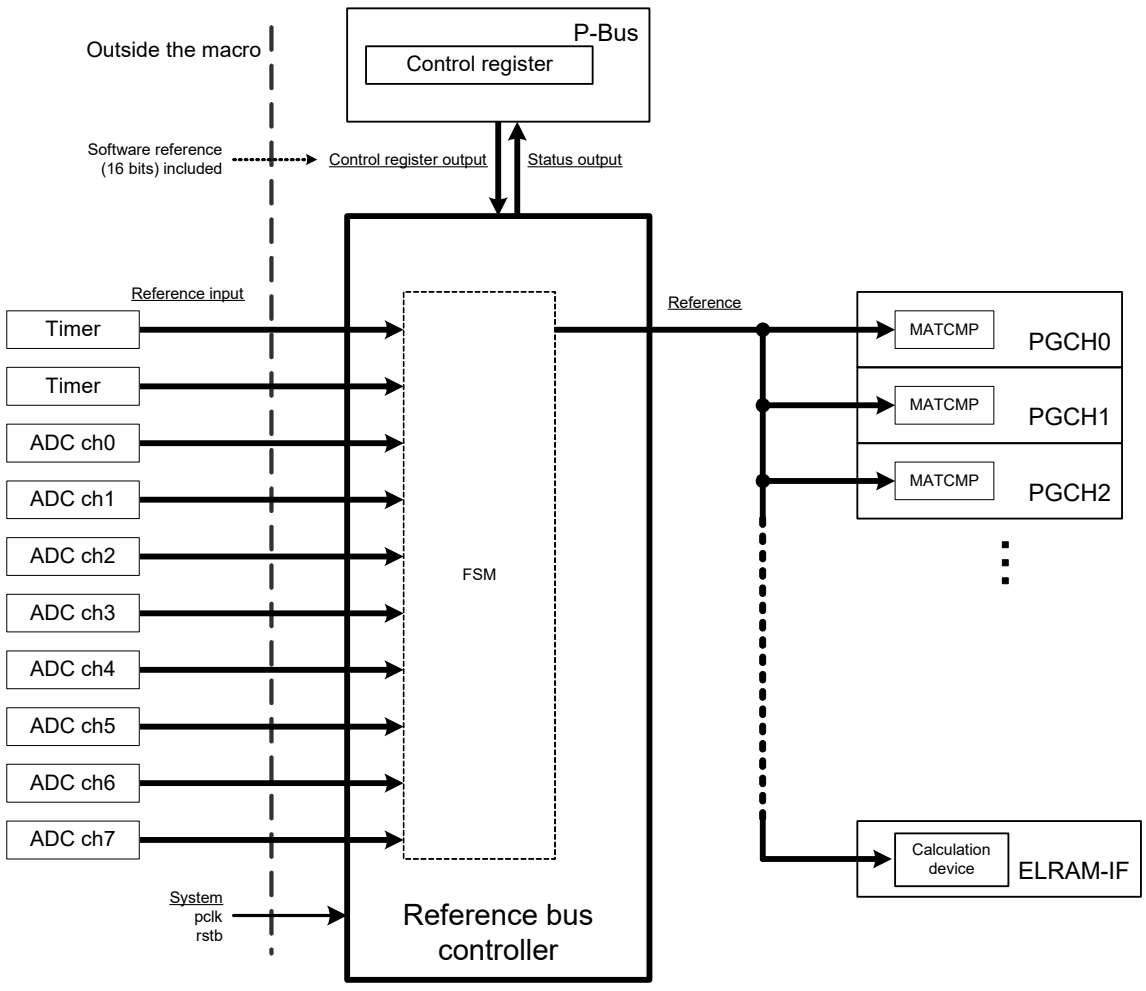
22.5.2.1 Features and Overview of Functions

External reference inputs (and software references) are multiplexed by time division within the APA.

The internal pulse generation channels (PGCH0 to PGCH15) are able to read desired references.

Reference outputs are initialized with the reset input signal PRESETZ or the module operation enable bit APAACMENAPA.

Figure 22.6 shows the circuit configuration.



Note 1. Reference inputs can be selected using the APA input selector. Refer to **Section 22.5.8, APA Input Selector**.

Figure 22.6 Configuration of Reference Control Circuit

22.5.2.2 Operation

All the reference inputs are multiplexed by time division with the following timing. Hereafter in this section, this time division processing is called “reference time division” and the unit time of time division (pclk * 1) is called a “reference slot”. By periodically assigning reference inputs to the reference slots, multiple references of different resolutions can be multiplexed.

Time division of the reference bus is controlled with the timing shown in **Figure 22.7**.

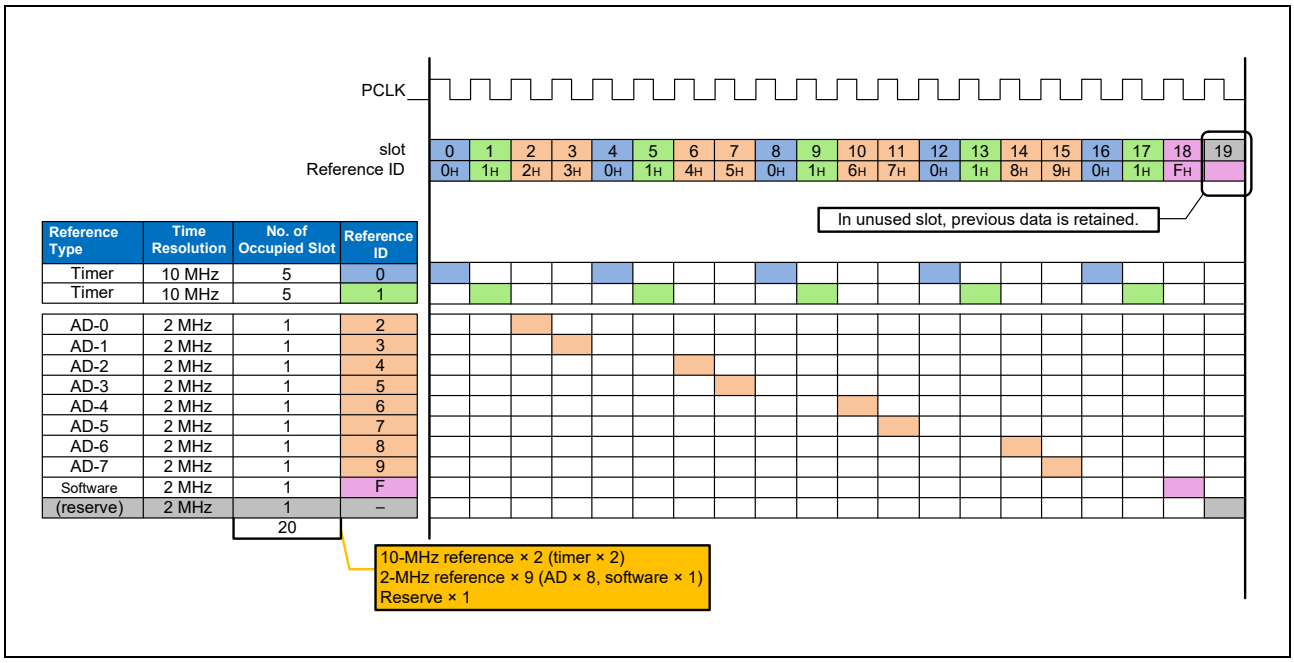


Figure 22.7 Reference Bus Operation

Table 22.48 shows the correspondence between available references and IDs.

Table 22.48 Types of Reference Inputs

Reference ID	Name	Time Resolution	Relative Designation	Reference Frequency Example at pclk=40 MHz
0_H	Reference input 0	pclk*4	Available	10 MHz
1_H	Reference input 1	pclk*4	Available	10 MHz
2_H	Reference input 2	pclk*20	Not available	2 MHz
3_H	Reference input 3	pclk*20	Not available	2 MHz
4_H	Reference input 4	pclk*20	Not available	2 MHz
5_H	Reference input 5	pclk*20	Not available	2 MHz
6_H	Reference input 6	pclk*20	Not available	2 MHz
7_H	Reference input 7	pclk*20	Not available	2 MHz
8_H	Reference input 8	pclk*20	Not available	2 MHz
9_H	Reference input 9	pclk*20	Not available	2 MHz
F_H	Software reference	pclk*20	Available	2 MHz

Note 1. A signal to be connected as a reference input must have slower update cycles than the time resolutions in **Table 22.48**. If it is faster, data will be lost in the reference control circuit, resulting in an unintended operation. For the reference inputs 0 and 1, update cycle can be adjusted with the bit select function of APA input selector described in **Section 22.5.8**.

22.5.3 Event Control

22.5.3.1 Features and Function Overview

The following processing is applied to external event inputs (and internal event inputs):

- Event inputs are selected for internal use. Out of 58 channels of the external event inputs and 6 channels of software events, 16 channels can be selected (these are hereinafter called selected event inputs). PWM outputs (16 channels) generated internally can also be used as event inputs (hereinafter called fixed event inputs).
- Selected event inputs (16 channels) and fixed event inputs (16 channels) are shared on a bus by the internal pulse generation channels (PGCH0 to PGCH15) and the element RAM (ELRAM).
- Time division control signals are generated so that the pulse generation channels (PGCH0 to PGCH15) can share common resources such as the element RAM and the calculation device.
- Event bus outputs are initialized with the reset input signal PRESETZ and the module operation enable bit APAACMENAPA.

Figure 22.8 shows the configuration of the event control circuit.

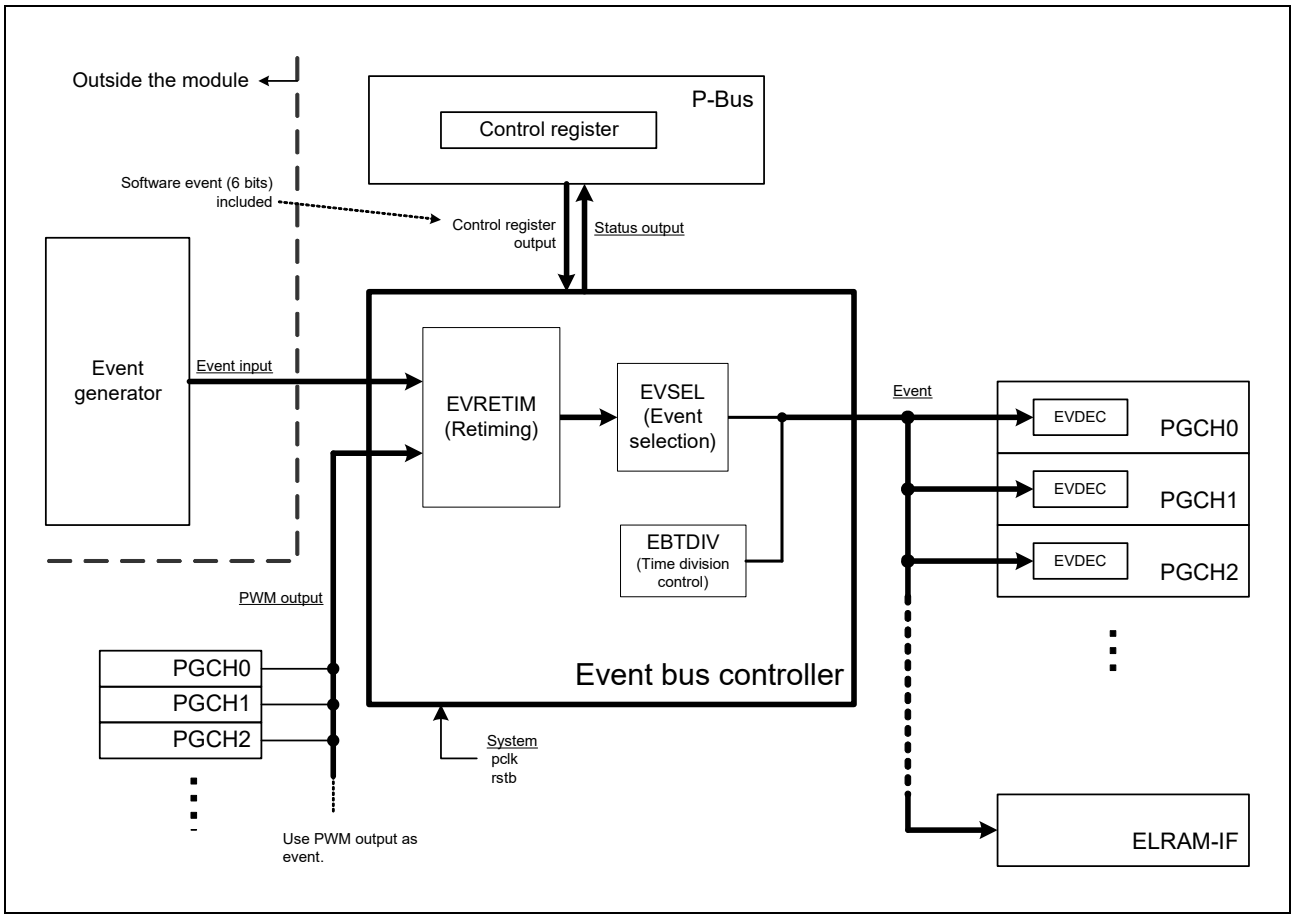


Figure 22.8 Configuration of Event Control Circuit

22.5.3.2 Operation: Selection of Event Inputs

The PWM outputs (16 channels) generated within the APA can be used as events by the pulse generation channels.

Among the 64 channels of the external event inputs (58 channels) and software events (6 channels), 16 channels can be used as events by the pulse generation channels.

Table 22.49 lists the inputs that the event bus can receive.

Table 22.49 Correspondence between Event Inputs and Event Buses

Event Type	Signal Source	Time Resolution	No of Channels	Corresponding Event Bus Bit
Fixed event input	PWM output: PWM outputs 0 to 15 (generated within APA)	pclk * 1	16	eb_dat_r[15:0]
Selected event input	External event input: External event sources 0 to 57, or software event: 6 channels → Selects any 16 channels from 64 channels	pclk * 1	16	eb_dat_r[31:16]

Note 1. A signal to be used as an event input must be slower than the time resolutions in **Table 22.49**. If it is faster, the APA cannot process it as an event input inside, resulting in data losses.

22.5.3.3 Operation: Generation of Time Division Control Signals

An event input is handled as a trigger of an exception processing on a pulse generation channel, in which access to the element RAM and various calculations take place. The APA performs the time division control to assign time to be used by the circuits in the APA (pulse generation channels and P-Bus) for sharing hardware resources such as the element RAM and the calculation device. Hereafter in this section, this time division control is called “event time division”, and a unit (pclk * 1) of time division is called an “event slot”. The event time division works with a cycle of 20 event slots (pclk * 20), which is defined as a “shared cycle”.

The time division control signals are generated with the timing shown in **Figure 22.9**.

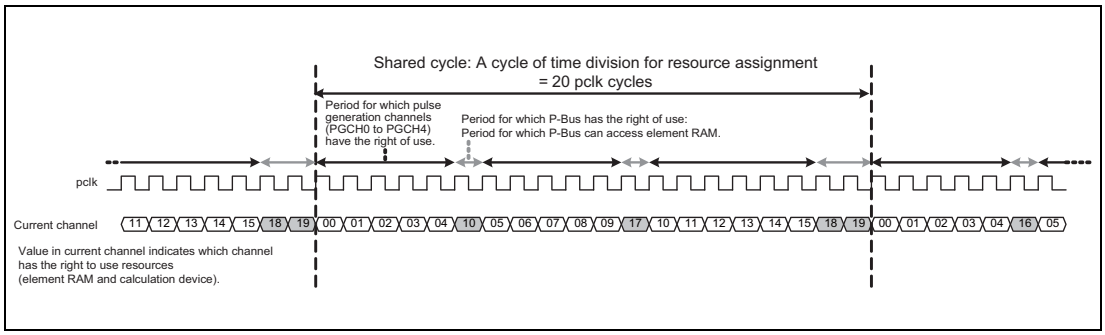


Figure 22.9 Operation of Event Time Division

Based on the current channel value, the APA assigns the access rights to the shared resources (element RAM and calculation device) as shown in **Table 22.50**.

Table 22.50 List of Current Channels and Shared Resource Assignment

Current Channel Value	Meaning	Current Channel Value	Meaning
00_H	PGCH0 can access shared resources.	09_H	PGCH9 can access shared resources.
01_H	PGCH1 can access shared resources.	11_H	P-Bus can access element RAM.
02_H	PGCH2 can access shared resources.	$0A_H$	PGCH10 can access shared resources.
03_H	PGCH3 can access shared resources.	$0B_H$	PGCH11 can access shared resources.
04_H	PGCH4 can access shared resources.	$0C_H$	PGCH12 can access shared resources.
10_H	P-Bus can access element RAM.	$0D_H$	PGCH13 can access shared resources.
05_H	PGCH5 can access shared resources.	$0E_H$	PGCH14 can access shared resources.
06_H	PGCH6 can access shared resources.	$0F_H$	PGCH15 can access shared resources.
07_H	PGCH7 can access shared resources.	12_H	P-Bus can access element RAM.
08_H	PGCH8 can access shared resources.	13_H	P-Bus can access element RAM.

22.5.4 Pulse Generation

22.5.4.1 Features and Overview of Functions

The following processing is applied to outputs of the reference bus and the event bus:

- A pulse generation channel extracts desired reference signals from the reference bus to generate a PWM based on the pulse generation conditions. Pulse generation conditions are expressed in elements.
- A pulse generation channel extracts desired event signals from the event bus, and performs exception processing triggered by these events. Conditions for generating exception processing from events are also expressed by elements.
- Exception processing of different pulse generation channels operates with time division.
- The PWM and INT outputs are initialized using the reset input signal presetz or the pulse generation channel operation enable bits APPACMENPGCH[15:0].

Figure 22.9 shows the configuration of the pulse generation circuit.

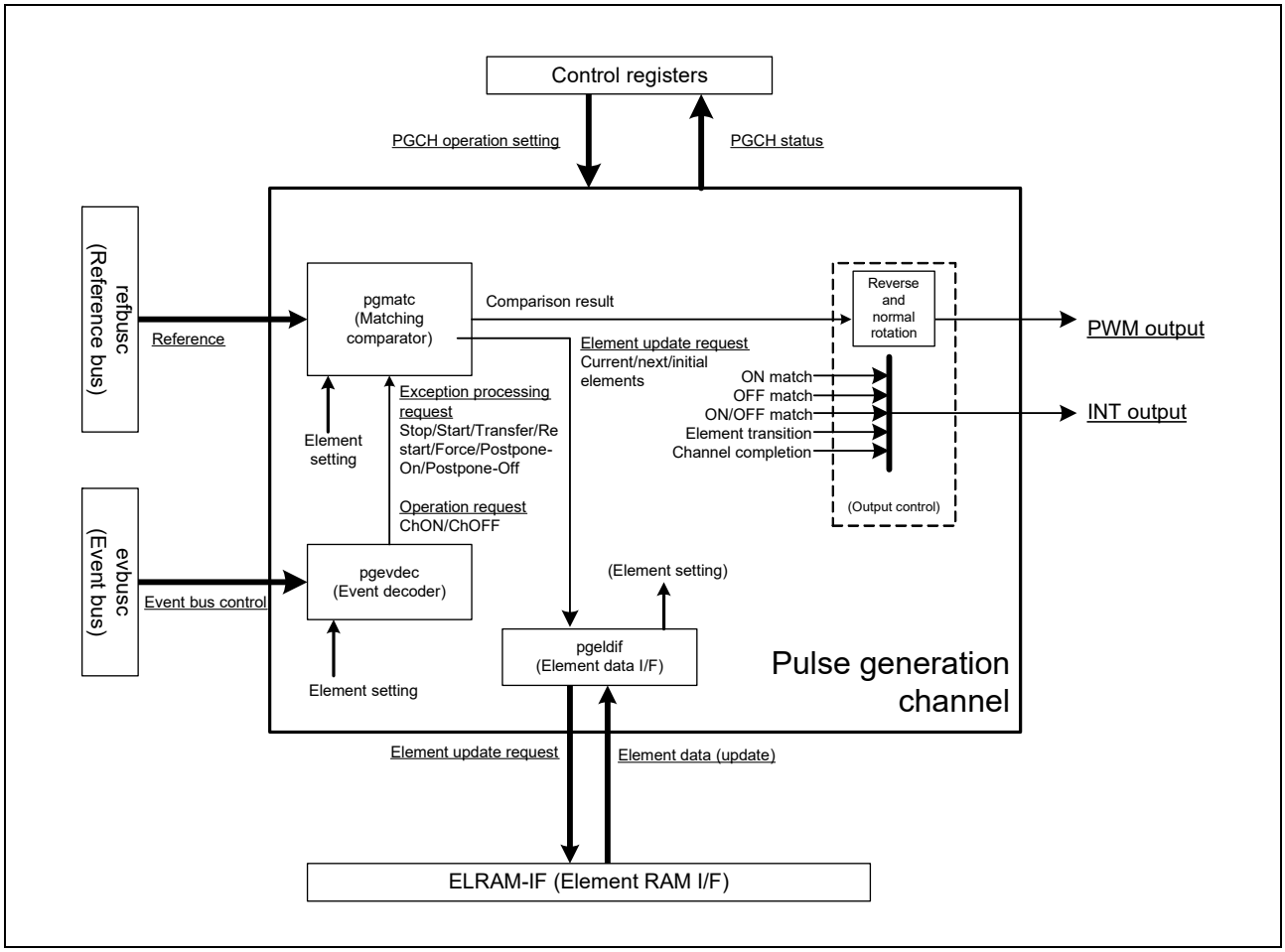


Figure 22.10 Configuration of Pulse Generation Circuit

22.5.4.2 Operation: Matching Comparator

(1) Overview

The matching comparator generates pulses and interrupts by comparing reference inputs with the matching conditions.

Matching comparison is performed according to the parameters defined in the element and the pulse generation channel. Upon a match detection, the pulse generation channel setting (see **22.4.5.1, APAA0CCGAn — APAA0 Channel Setting Registers An (n = 00 to 15)**) and the element setting (see **22.4.6.1, APAA0ELMAn — APAA0 Element Setting Registers An (n = 000 to 127)**) to **22.4.6.3, APAA0ELMCn — APAA0 Element Setting Registers Cn (n = 000 to 127)**) determine issuing of an element update request to the element control module (element RAM-IF).

Table 22.51 PWM Outputs and Interrupt Outputs

Type	Time Resolution	Number of Channels	Minimum Pulse Width	Remarks
PWM output	pclk * 1	16	pclk * 40 (high) pclk * 40 (low)	The minimum pulse width shown here is applicable when ON/OFF occurs on the match condition specified by an element. If the match status is forcibly changed by exception processing, a shorter pulse width may result from the conflict with a match.
Interrupt output	pclk * 1	16	pclk * 1 (high)	Only one-shot pulse (rise) can be output.

Note 1. PWM output matching conditions can be specified by an element. A matching condition must satisfy the minimum pulse widths listed in the above table (see **22.5.7.1, Matching Condition Setting 1**). Specifying a matching condition with a pulse width of pclk*40 or shorter might result in an unintended pulse width.

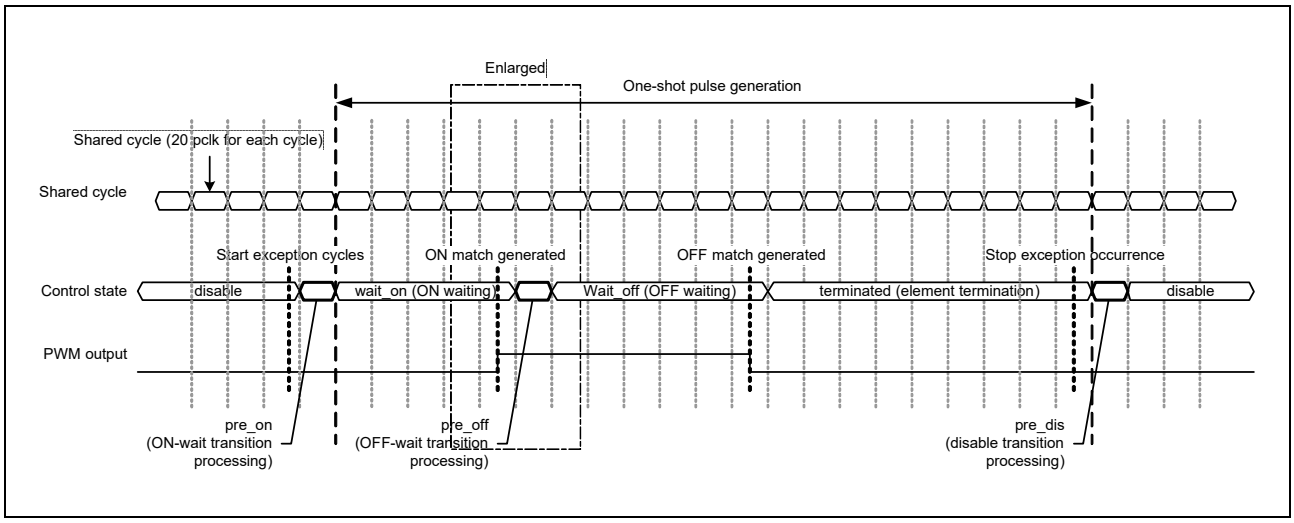


Figure 22.11 Operation of Matching Comparator

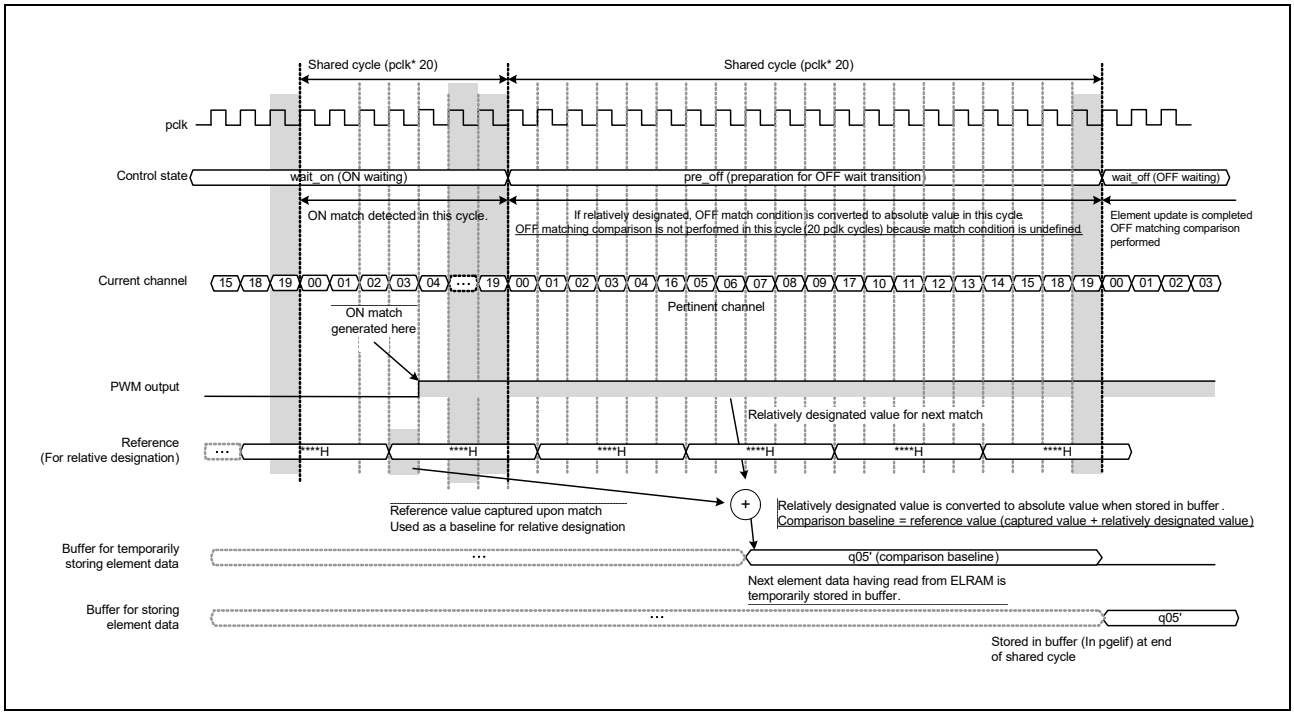


Figure 22.12 Operation of Matching Comparator (Enlarged View)

(2) State Control

The matching comparator controls operation starting/stopping, ON matching comparison, OFF matching comparison, element sequence according to the state control rules below.

- States are controlled every shared cycle (pclk * 20) based on the current channel signal generated from the event bus.
- State control is linked with the matching control described in **(3) Matching control** in this section.
- States of the matching comparator are controlled as shown in the state transition shown in **Figure 22.13**.
- The states shown in **Figure 22.13** include sub-states. For the relationships between the states and sub-states, refer to **Table 22.52** and **Figure 22.13** to **Figure 22.21**.

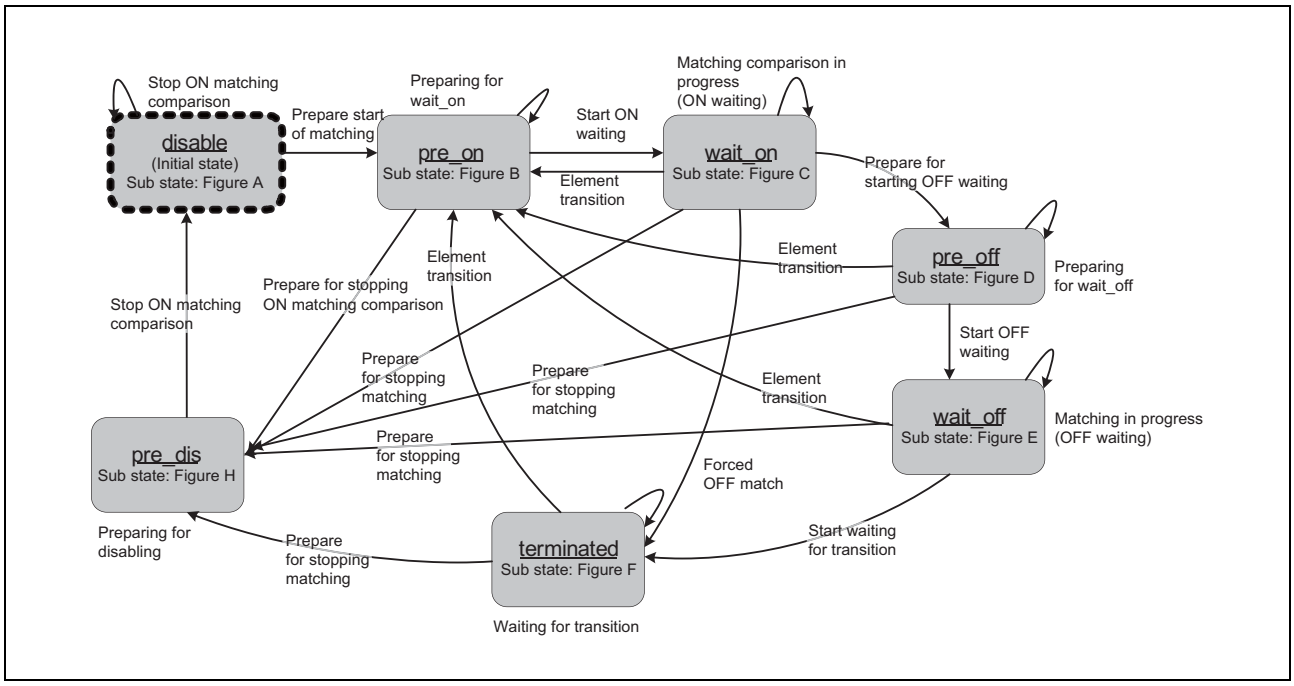


Figure 22.13 State Transition of Matching Comparator

In the matching comparison state, the operation proceeds as described in **Table 22.52**.

Table 22.52 List of States (1/2)

State			Input Condition (left: high priority; right: low priority)							
State	Sub-state	Operation Overview	Channel ON/OFF	Stop exception	Restart exception	Transfer exception	Force exception	Match	Postpone exception	Start exception
disable	disable	State of complete stop, where no matching comparison is made. The initial state of the state machine.	√ (ON only)	×	×	—	—	—	—	×
	dis_wstart	State of matching comparison stop. Operation permission = enabled, and Start/Restart is awaited.	√ (OFF only)	×	√	—	—	—	—	√
pre_on	pre_on_cur	Preparing for transition to ON waiting. Element update (current element) is performed. Matching comparison is not performed. Used when looping in element continuation.	√ (OFF only)	√	√	√	√	—	×	×
	pre_on_init	Preparing for transition to ON waiting. Element update (initial element) is performed. Matching comparison is not performed	√ (OFF only)	√	√	—	—	—	—	×
	pre_on_nxt	Preparing for transition to ON waiting. Element update (next element) is performed. Matching comparison is not performed	√ (OFF only)	√	√	—	—	—	—	×
wait_on	waiton	For ON matching comparison, a match or a Transfer exception input is awaited.	√ (OFF only)	√	√	√	√	√	√	×
	waiton_tron	For On matching comparison, a Transfer has been input. A match is awaited.	√ (OFF only)	√	√	√	√	√	√	×
	forcedoff_troff	For ON matching comparison, a forced OFF match has occurred by a Force exception. A Transfer exception input is awaited.	√ (OFF only)	√	√	√	√	× (OFF match occurs)	√	×
	on_postponed	For ON matching comparison, a Postpone exception has been input. A match occurrence and release from the Postpone state are awaited. ON matching comparison is operating during the Postpone input period, but the PWM pin is not affected, and a state transition is not initiated until release from the Postpone state.	√ (OFF only)	√	√	√	√	√	√	×
pre_off	pre_off	Preparing for transition to OFF waiting. Element update (current element) is performed. Matching comparison is not performed	√ (OFF only)	√	√	√	√	—	×	×

Table 22.52 List of States (2/2)

State			Input Condition (left: high priority; right: low priority)							
State	Sub-state	Operation Overview	Channel ON/OFF	Stop exception	Restart exception	Transfer exception	Force exception	Match	Postpone exception	Start exception
wait_off	waitoff	For OFF matching comparison, a match or a Transfer exception input is awaited.	√ (OFF only)	√	√	√	√	√	√	×
	waitoff_tron	For OFF matching comparison, a Transfer has been input. A match is awaited.	√ (OFF only)	√	√	√	√	√	√	×
	matoff_troff	For OFF matching comparison, a Transfer has not been input and an OFF match has occurred. A Transfer exception is awaited.	√ (OFF only)	√	√	√	√	× (OFF match occurs)	√	×
	off_postponed	For OFF matching comparison, a Postpone exception has been input. A match occurrence and release from the Postpone state are awaited. OFF matching comparison is operating during the Postpone input period, but the PWM pin is not affected and a state transition is not initiated until release from the Postpone state.	√ (OFF only)	√	√	√	√	√	√	×
terminate	terminate	There is no next element to be executed (i.e. the last element has been completed). Matching comparison is not performed, and a Stop or Restart exception is awaited.	√ (OFF only)	√	√	—	—	—	—	×
pre_dis	pre_dis	Preparing for transition to “disable”. The internal status registers are reset (the channel element settings are retained).	×	×	×	—	—	—	—	×

Note: √: Processing is possible.
×: Processing is impossible (occurrence of an input condition is ignored).
—: Processing is impossible (no effective input conditions can occur since definition is impossible in the pertinent state).

For details of the input conditions, refer to the related sections listed below.

- Channel ON/OFF: **(5) Channel ON/OFF** in **Section 22.5.4.2, Operation: Matching Comparator**
- Stop exception: **(1) Stop Exception** in **Section 22.5.4.3, Operation: Event Decoder**
- Restart exception: **(3) Restart Exception** in **Section 22.5.4.3, Operation: Event Decoder**
- Transfer exception: **(4) Transfer Exception** in **Section 22.5.4.3, Operation: Event Decoder**
- Force exception: **(5) Force Exception** in **Section 22.5.4.3, Operation: Event Decoder**
- Matching comparison: **(3) Matching control** in **Section 22.5.4.2, Operation: Matching Comparator**
- Postpone exception: **(6) Postpone Exception** in **Section 22.5.4.3, Operation: Event Decoder**

- Start exception: **(2) Start Exception** in **Section 22.5.4.3, Operation: Event Decoder**

The input conditions shown in **Table 22.52** are enabled at the points in time as shown in **Figure 22.14**.

When more than one input condition is detected, they are processed according to the priority shown in **Table 22.52**.

For details of the input conditions, refer to **Section 22.5.4.3, Operation: Event Decoder**.

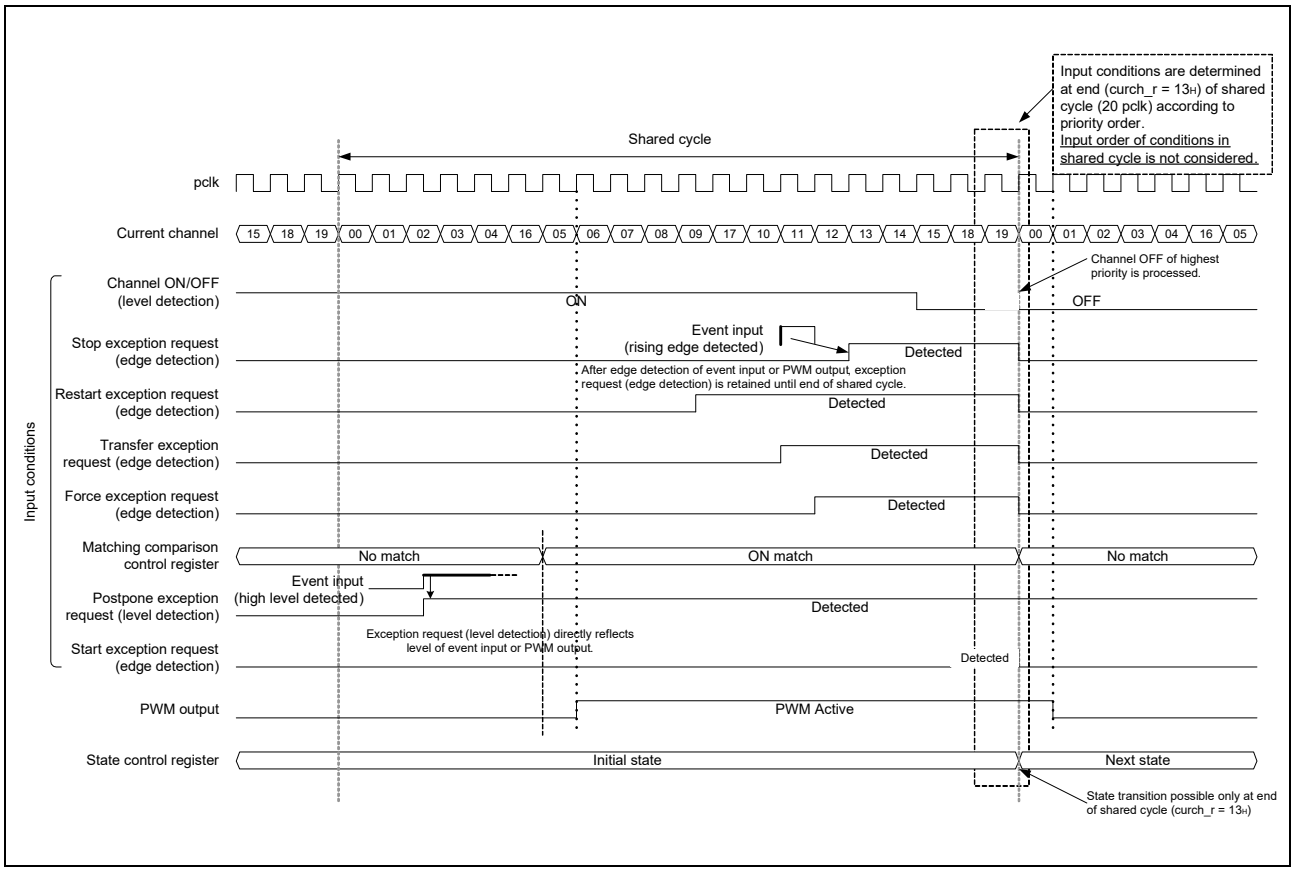


Figure 22.14 Input Condition Determination Timing

Figure 22.15 to **Figure 22.21** show details of each state of the matching comparator.

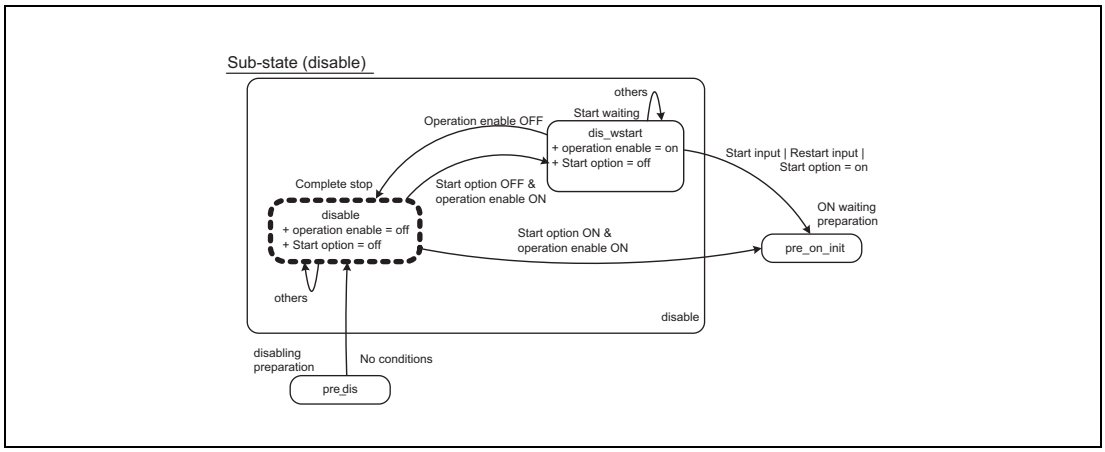


Figure 22.15 State Transition of Matching Comparator (figure A indicated in Figure 22.13, State Transition of Matching Comparator)

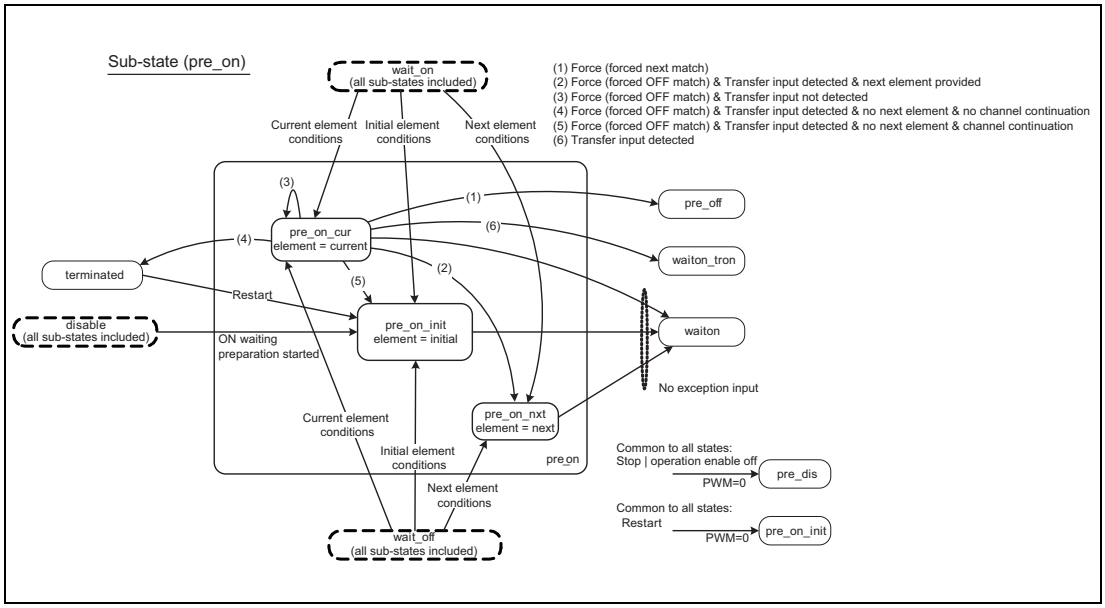


Figure 22.16 Sub-State Transition in pre_on State (figure B indicated in Figure 22.13, State Transition of Matching Comparator)

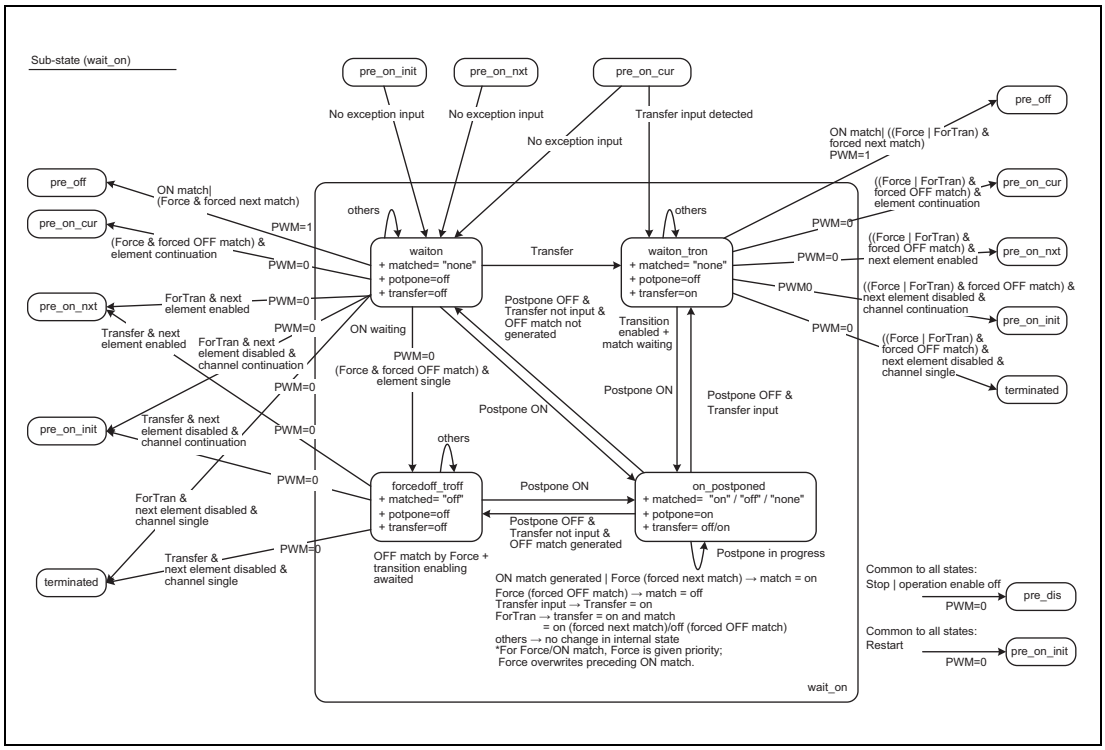


Figure 22.17 Sub-State Transition in wait_on State (figure C indicated in Figure 22.13, State Transition of Matching Comparator)

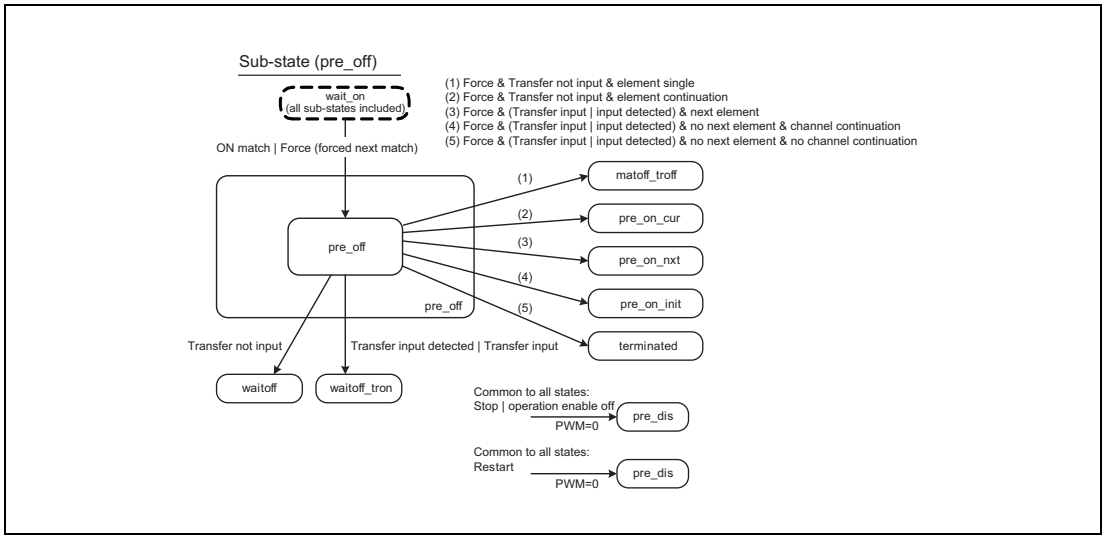


Figure 22.18 Sub-State Transition in pre_off State (figure D indicated in Figure 22.13, State Transition of Matching Comparator)

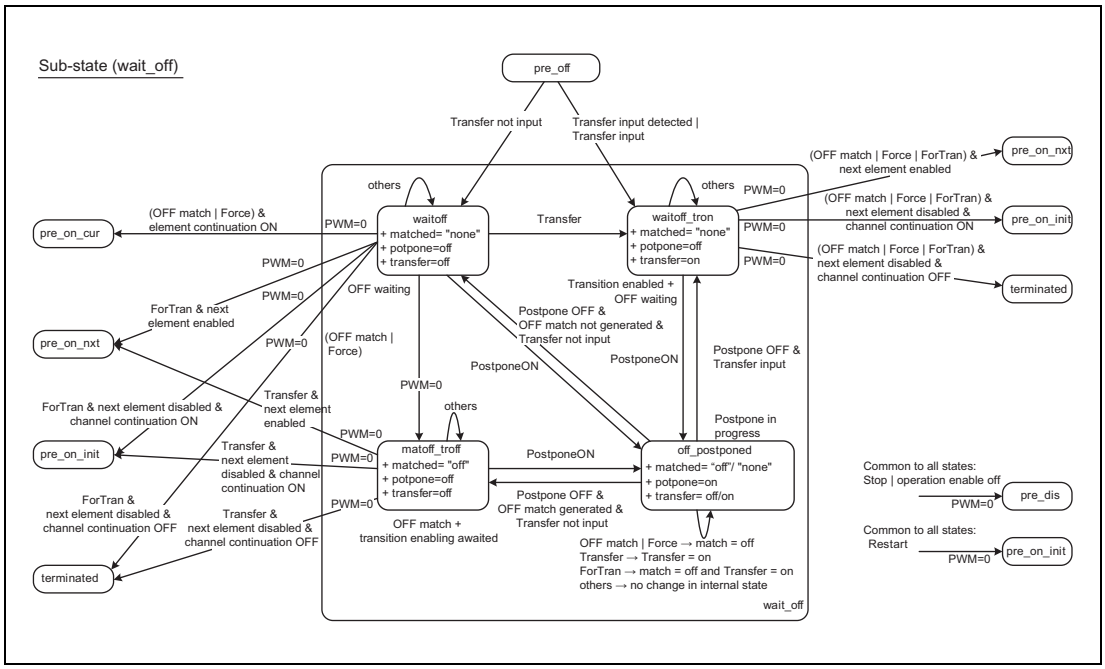


Figure 22.19 Sub-State Transition in wait_off State (figure E indicated in Figure 22.13, State Transition of Matching Comparator)

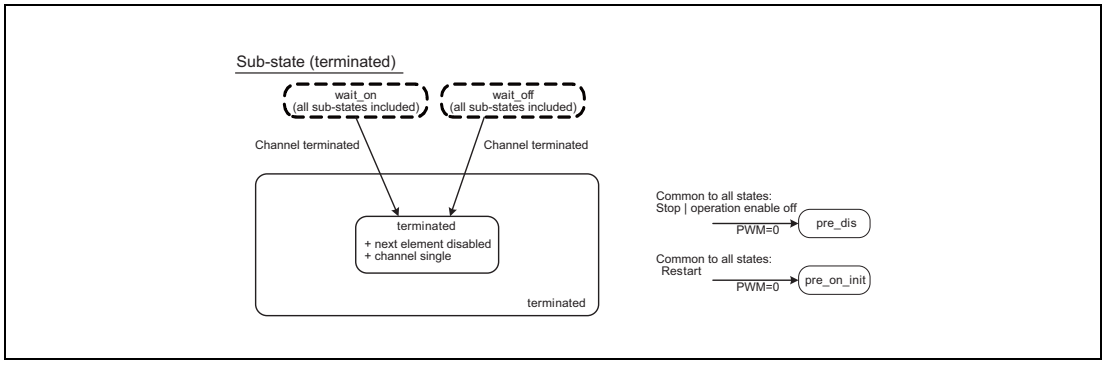


Figure 22.20 Sub-State Transition in Terminated State (figure F indicated in Figure 22.13, State Transition of Matching Comparator)

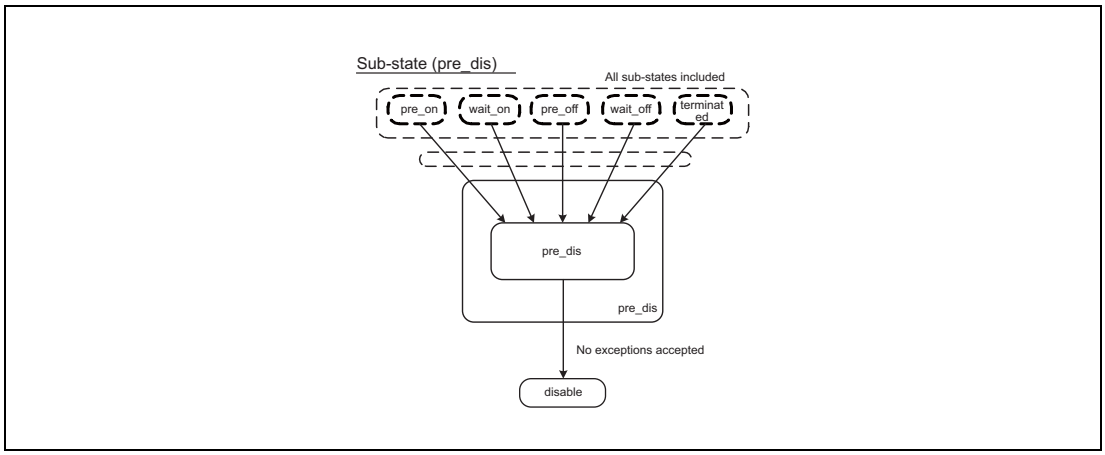


Figure 22.21 Sub-State Transition in pre_dis State (figure G indicated in Figure 22.13, State Transition of Matching Comparator)

(3) Matching control

Matching is controlled based on comparison of the reference inputs with the comparison conditions using the matching comparator. Operations are described in detail below.

- The outputs from the reference bus (reference inputs are time-division multiplexed) are compared to the conditions specified using the matching comparison registers (refer to **Table 22.53** and **Table 22.54**). If all the conditions agree with the reference bus outputs, a match detection is determined and the match control register settings are modified.
- Comparison is linked to state control: only ON matching comparison is performed in the wait_on system states, and only OFF matching comparison is performed in the wait_off system states. ON matching and OFF matching comparison are not performed simultaneously.
- A match is detected only once in a single state and cannot occur more than once.
- The matching comparison conditions are defined for each element.
- With using the Force exception, the match control status can be changed independently of ON matching and OFF matching comparison status. For details of the Force exception, refer to **(5) Force Exception** in **Section 22.5.4.3, Operation: Event Decoder**.

Table 22.53 ON Matching Comparison

Item		Description
Applicable states	4 wait_on system states	ON matching comparison is performed in the following states: waiton, waiton_tron, forcedoff_troff, on_postponed
Comparison condition 1 (reference number)	APAAELDRBIDON[3:0]	Reference number is specified. For the correspondence between the settings and the reference numbers, refer to **Table 22.37**.
Comparison condition 2 (large/small relationship)	APAAELDMCON[1:0]	Whether the reference value is larger, smaller, or equal to the comparison baseline is specified. In comparison of large and small values, the reference value is handled as unsigned. 00_B: [Reference value of specified reference number] == [comparison baseline] 01_B: [Reference value of specified reference number] > [comparison baseline] 10_B: [Reference value of specified reference number] < [comparison baseline]
Comparison condition 3 (comparison baseline)	APAAELDRBDATON[15:0]	Comparison baseline for ON matching comparison is specified as an unsigned value.

In ON matching comparison, when all of the three comparison conditions shown in **Table 22.53** are satisfied, an ON match is detected in the next clock edge. The control signals change as shown in **Table 22.54**.

Table 22.54 Signal Change upon ON Match Occurrence

Item		Description
Match control	APAAMATSTAT[1:0]	Match not detected (00_B) → ON match detected (01_B)
PWM output	APAAEOPWM[X] (X = channel number)	Inactive level → Active level • Active level is specified for each channel using chn_lev register. • Level does not change if ON match is detected in Postpone state.
slotID (register for holding timing information upon match occurrence)	el_slotid_r[4:0]	slotID (reference bus output) upon ON match occurrence is held.

By inputting the Force exception, an ON match detection can be caused independently of the above comparison conditions. For details, refer to **(5) Force Exception** in **Section 22.5.4.3, Operation: Event Decoder**.

Table 22.55 OFF Matching Comparison

Item		Description
Applicable states	4 wait_off system states	OFF matching comparison is performed in the following states: waitoff, waitoff_tron, matoff_troff, off_postponed
Comparison condition 1 (reference number)	APAAELDRBIDOFF[3:0]	Reference number is specified. For the correspondence between the settings and the reference numbers, refer to **Table 22.37**.
Comparison condition 2 (large/small relationship)	APAAELD_MCOFF[1:0]	Whether the reference value is larger, smaller, or equal to the comparison baseline is specified. In comparison of large and small values, the reference value is handled as unsigned. 00_B: [Reference value of specified reference number] == [comparison baseline] 01_B: [Reference value of specified reference number] > [comparison baseline] 10_B: [Reference value of specified reference number] < [comparison baseline]
Comparison condition 3 (comparison baseline)	APAAELDRBDATOFF[15:0]	Baseline for OFF matching comparison. It is specified as an unsigned value.

In OFF matching comparison, when all of the three comparison conditions shown in **Table 22.55** are satisfied, an OFF match is detected in the next clock edge. The control signals change as shown in **Table 22.56**.

Table 22.56 Signal Change upon OFF Match Occurrence

Item		Description
Match control	APAAMATSTAT[1:0]	ON match detected (01_B) → OFF match detected (10_B)
PWM output	APAAEOPWM[X] (X = channel number)	Active level → Inactive level • Active level is specified for each channel using chn_lev register. • Level does not change if OFF match is detected in Postpone state.
slotID (register for holding timing information upon match occurrence)	el_slotid_r[4:0]	slotID (reference bus output) upon OFF match occurrence is held.

By inputting the Force exception, an OFF match detection can be caused independently of the above comparison conditions. For details, refer to **(5) Force Exception** in **Section 22.5.4.3, Operation: Event Decoder**.

Figure 22.22 shows ON match control operations.

This figure is also applicable to OFF match control operations if the comparison conditions for ON match are replaced with those for OFF match.

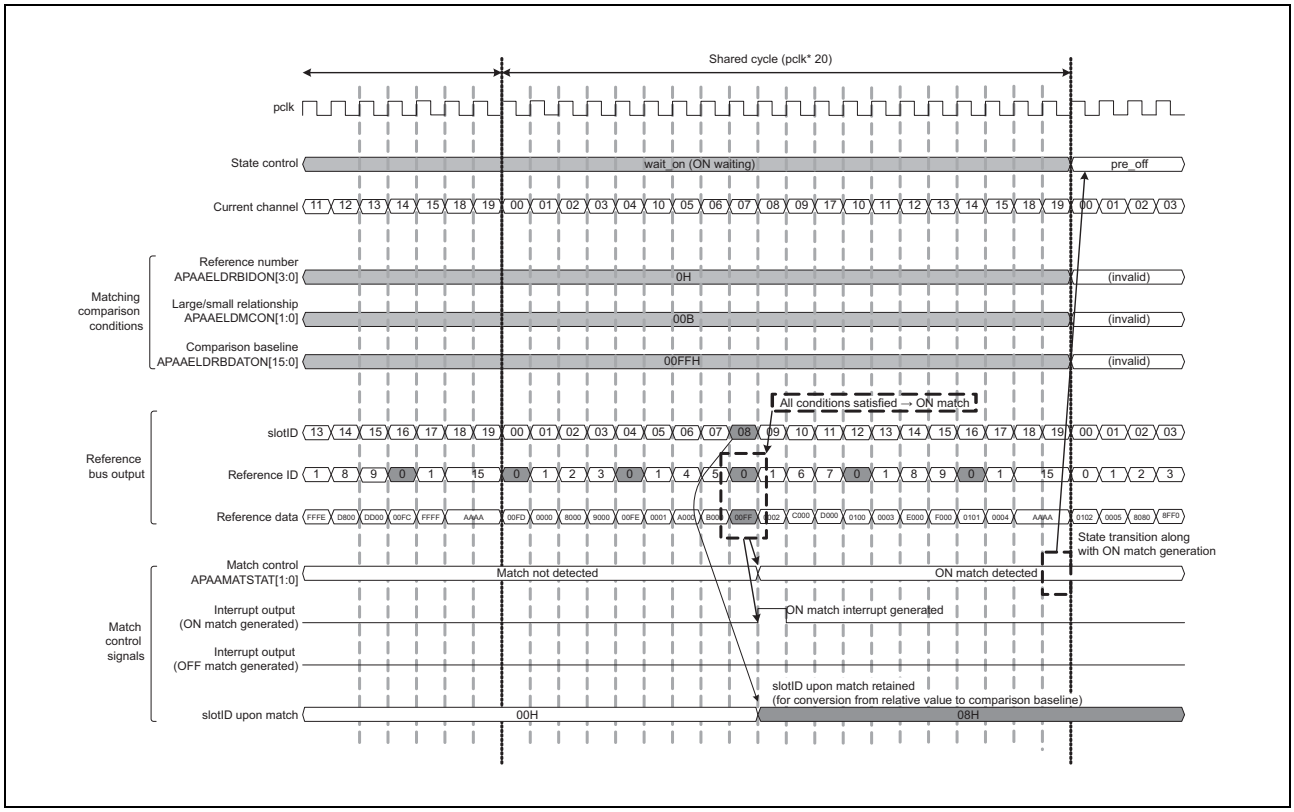


Figure 22.22 ON Match Control Operations

(4) Interrupts

While the matching comparator is operating, an interrupt can be generated on either of the conditions shown in **Table 22.57**.

Interrupts can be independently set using the APAA0CCGAn.APAACHNINTSEL[2:0] bits (n = 0 to 15) for each channel.

Interrupts are output as a one-shot active-high pulse.

Table 22.57 Interrupt Generation Conditions

APAACHNINTSEL[2:0] Setting	Interrupt Generation Condition	Interrupt Generation Timing
000_B	After element transition	An interrupt is generated at a shared cycle boundary after element update (state: pre_on). When element continuation is enabled (APAA0ELMAn.APAAELMCONT = 1), an interrupt is also generated at a shared cycle boundary after pre_on_cur state. Refer to **Figure 22.23**.
001_B	Upon channel operation completion	An interrupt is generated at a shared cycle boundary after last element completion (match or OFF match by Force exception). Refer to **Figure 22.24**.
010_B	Upon ON match	An interrupt is generated upon ON match (match by comparison or Force exception) during ON waiting (state: wait_on). Specifically, an interrupt is output in the following cases. • ON match detection by comparison (refer to **Figure 22.25**) • ON match detection by Force exception (refer to **Figure 22.26**) • Conflict between ON match detection by comparison and ON match detection by Force exception (refer to **Figure 22.27**) • Conflict between ON match detection by comparison and OFF match detection by Force exception (refer to **Figure 22.28**) An interrupt is output also in Postpone state whether ON match occurs or not. Refer to **Figure 22.46**.
011_B	Upon OFF match	An interrupt is generated upon OFF match (match by comparison or Force exception) during OFF waiting (state: wait_off) or upon forced OFF match by Force exception during ON waiting (state: wait_on). Refer to the same figures listed in the description of “Upon ON” description above. As in “Upon ON Match” above, an interrupt is output also in Postpone state.
100_B	Upon either ON match or OFF match	An interrupt is generated simultaneously with PWM assertion/negation upon either ON or OFF match. Refer to the same figures listed in the description of “Upon ON Match” above. As in “Upon ON Match” above, an interrupt is output also in Postpone state.

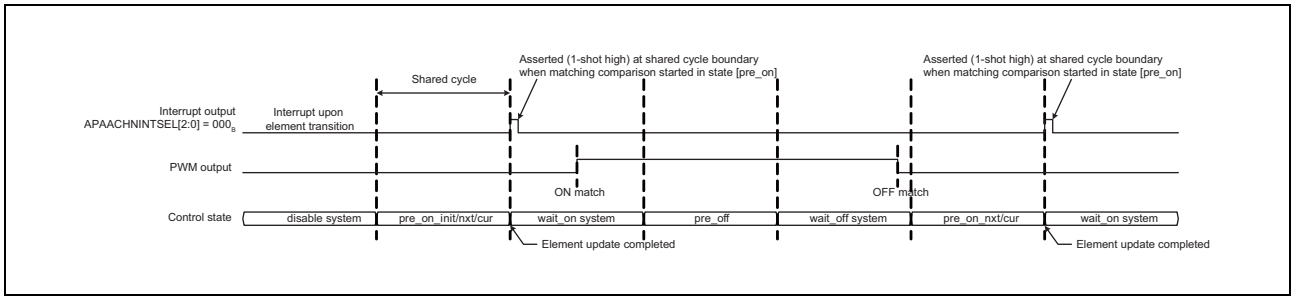


Figure 22.23 Interrupt Generation (after element transition)

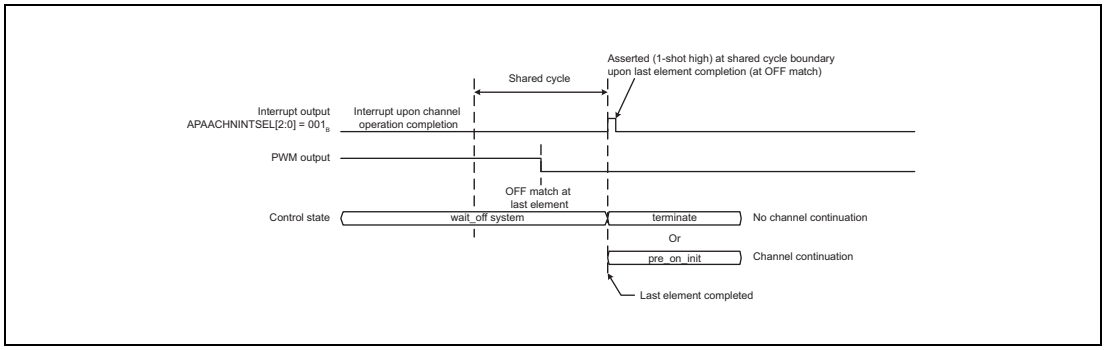


Figure 22.24 Interrupt Generation (upon channel operation completion)

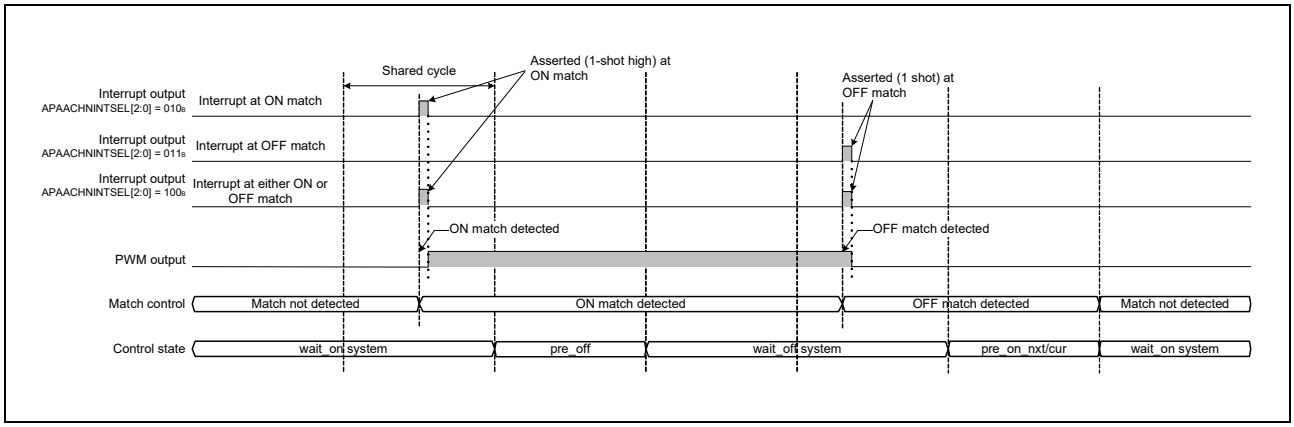


Figure 22.25 Interrupt Generation (upon ON/OFF/ON+OFF match by comparison)

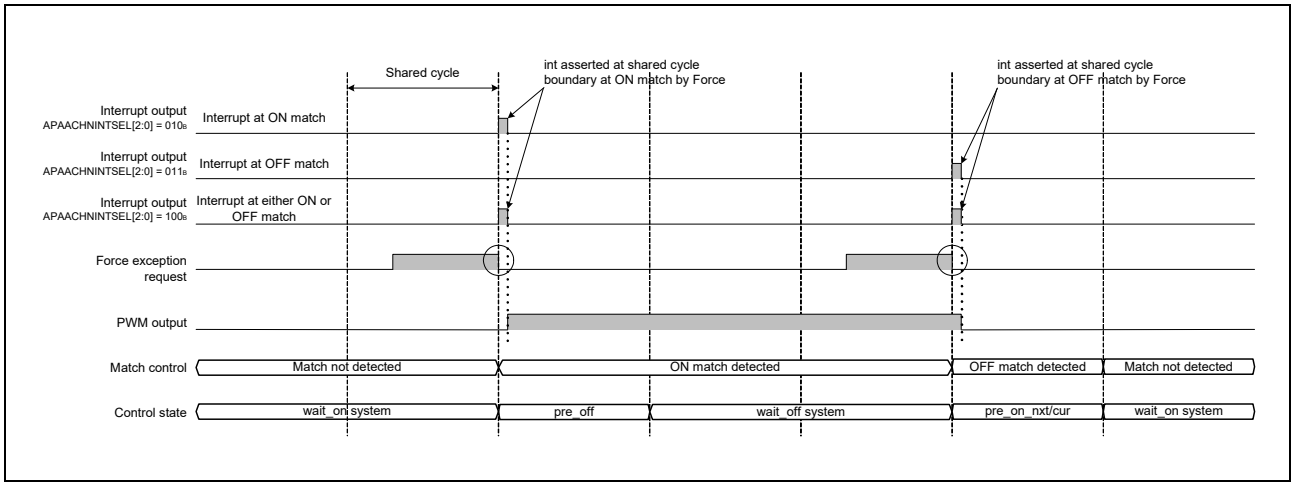


Figure 22.26 Interrupt Generation (upon ON/OFF/ON+OFF match by Force exception)

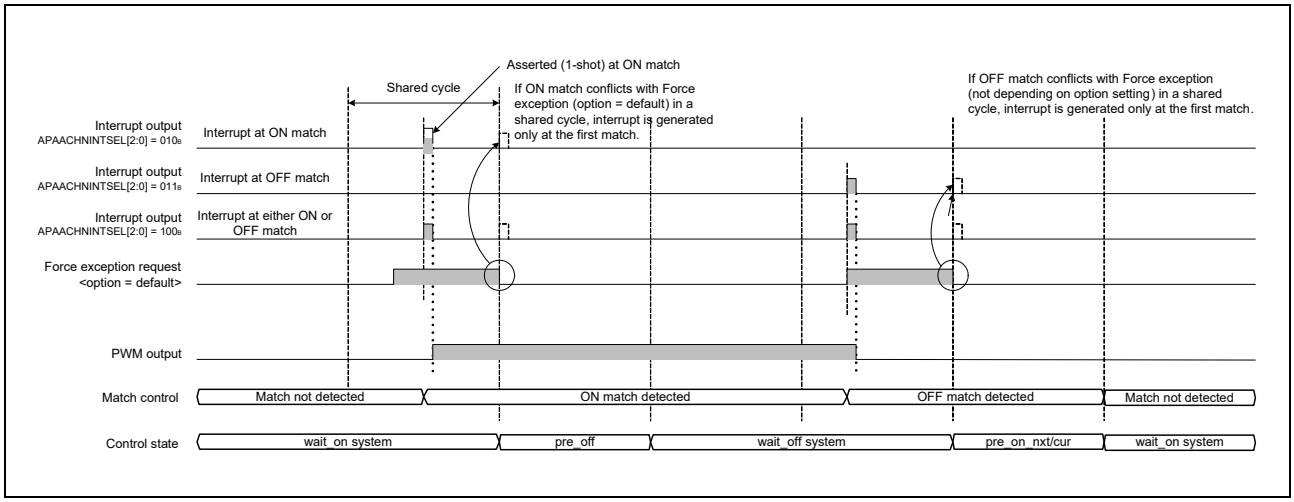


Figure 22.27 Interrupt Generation (upon conflict between match detection by comparison and match detection by Force exception 1/2)

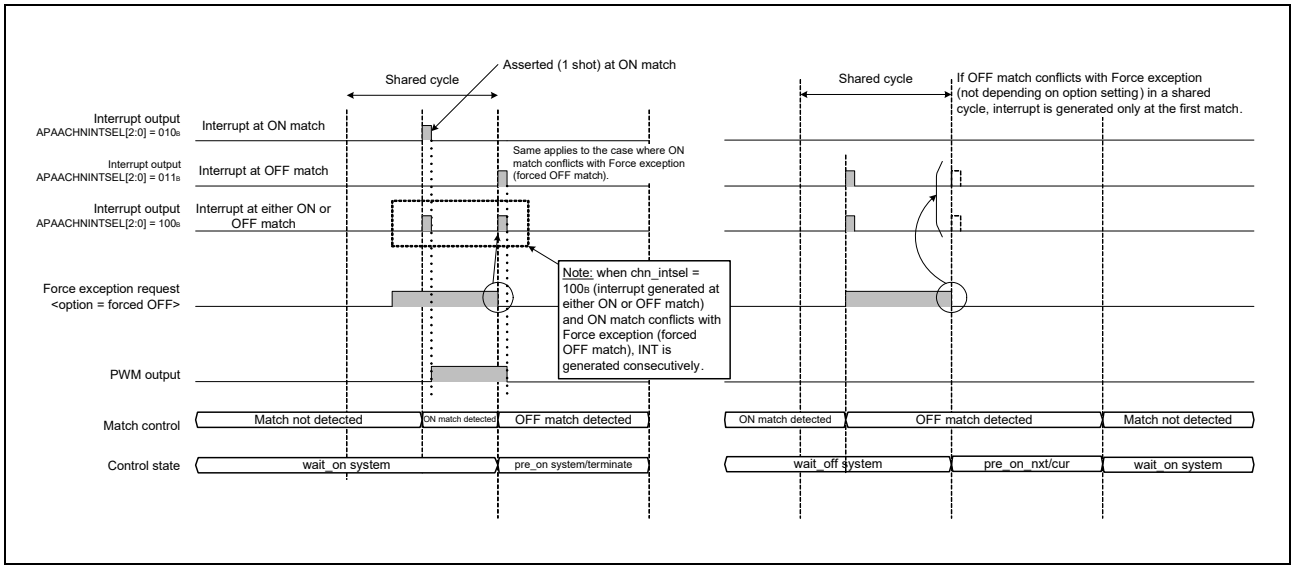


Figure 22.28 Interrupt Generation (upon conflict between match detection by comparison and match detection by Force exception 2/2)

(5) Channel ON/OFF

Channel ON/OFF is an input condition according to the exception processing requests generated by the event decoder, which are described in **Section 22.5.4.3, Operation: Event Decoder**. Channel on/off operations cause the operations shown in **Table 22.57**.

Table 22.58 Channel ON/OFF Operations

Item	Description
Operation (channel ON)	• Channel ON operation is used as the condition input to start the matching comparator in the stop state. • Channel ON operation is valid only in the disable state; it is ignored in the other states. • Channel ON operation does not cause changes in PWM output or interrupt output.
Operation (channel OFF)	• Channel OFF operation is used as the condition input to stop the matching comparator in the operating state. • Channel OFF operation is valid while the channel is operating (in the states other than disable and pre_dis). • Channel OFF operation forcibly causes PWM output to become inactive one pclk cycle after state transition. Here, no interrupt is output.
Assignment	• APAA0CHEN[15:0] register is used for setting. Each bit corresponds to a channel.
Detection type	• Level detection (active high only; active level cannot be selected)
Option	• None
Note	• None

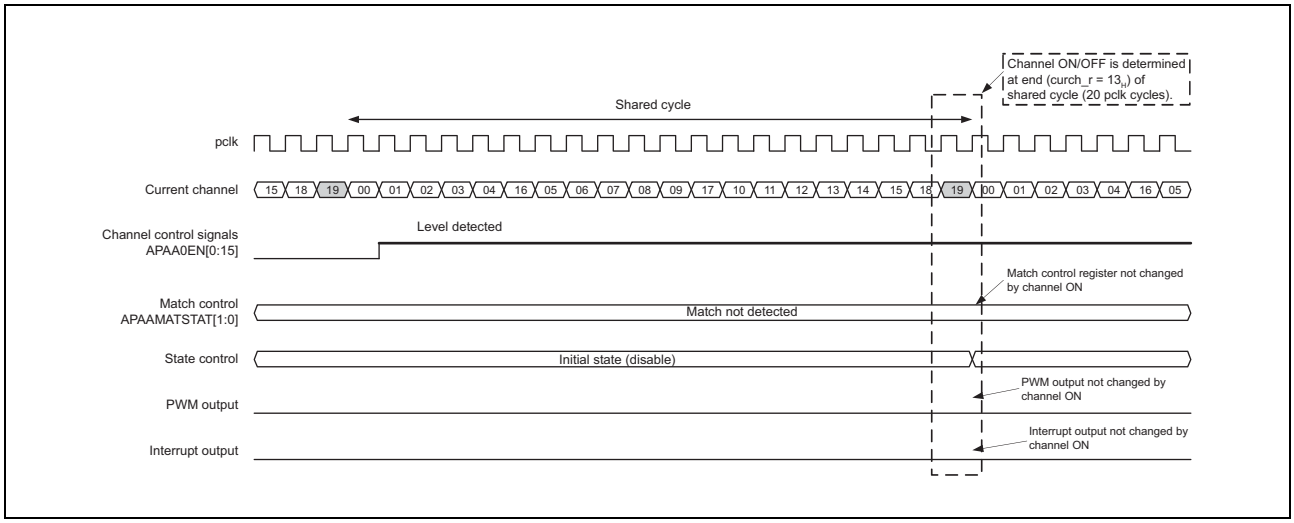


Figure 22.29 Channel ON Acceptance Timing

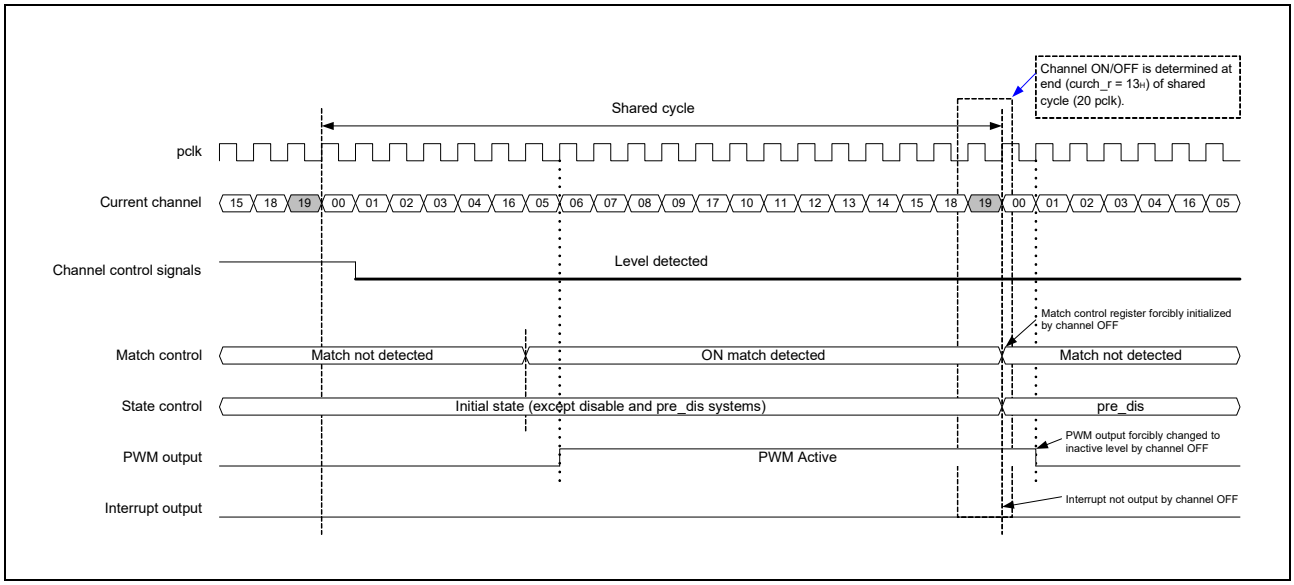


Figure 22.30 Channel OFF Acceptance Timing

22.5.4.3 Operation: Event Decoder

The event decoder generates exception processing requests based on the event bus or PWM outputs generated by the pulse generation channels.

Figure 22.31 and **Figure 22.32** are timing charts of the exception signals.

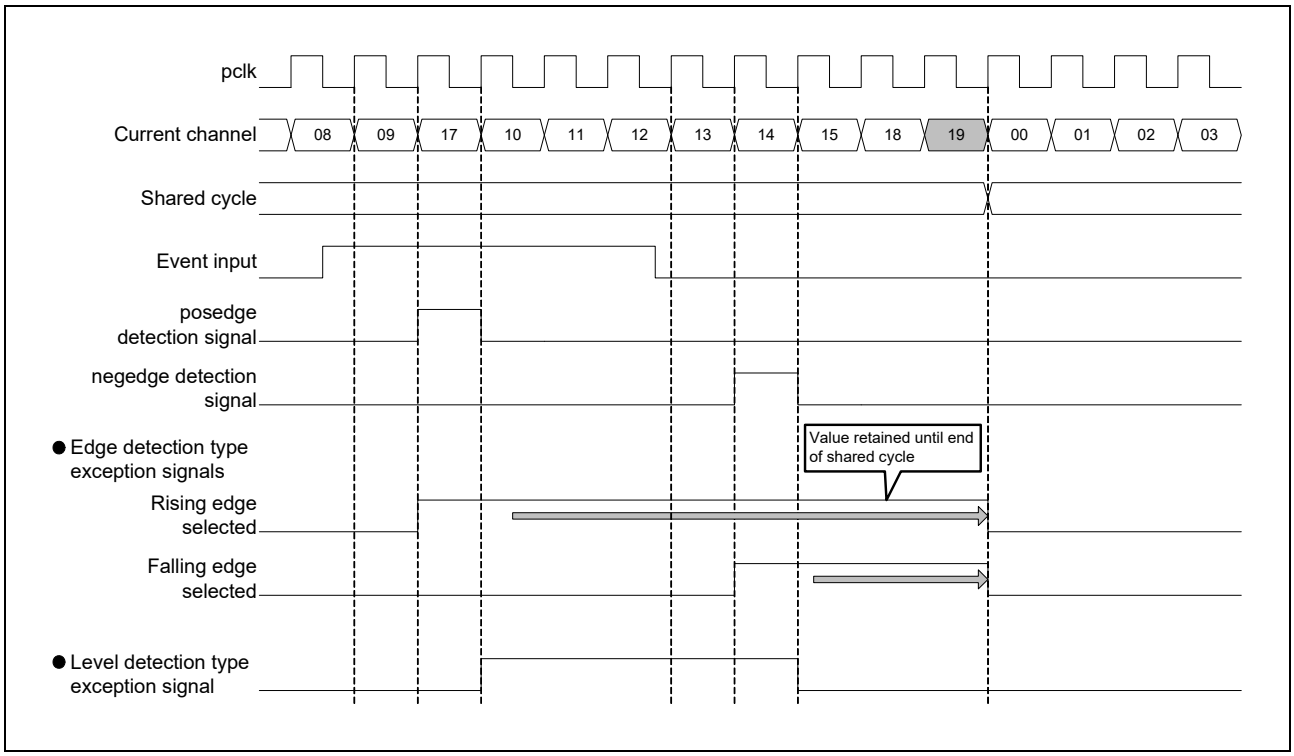


Figure 22.31 Timing Chart of Exception Signals when External Event is Selected

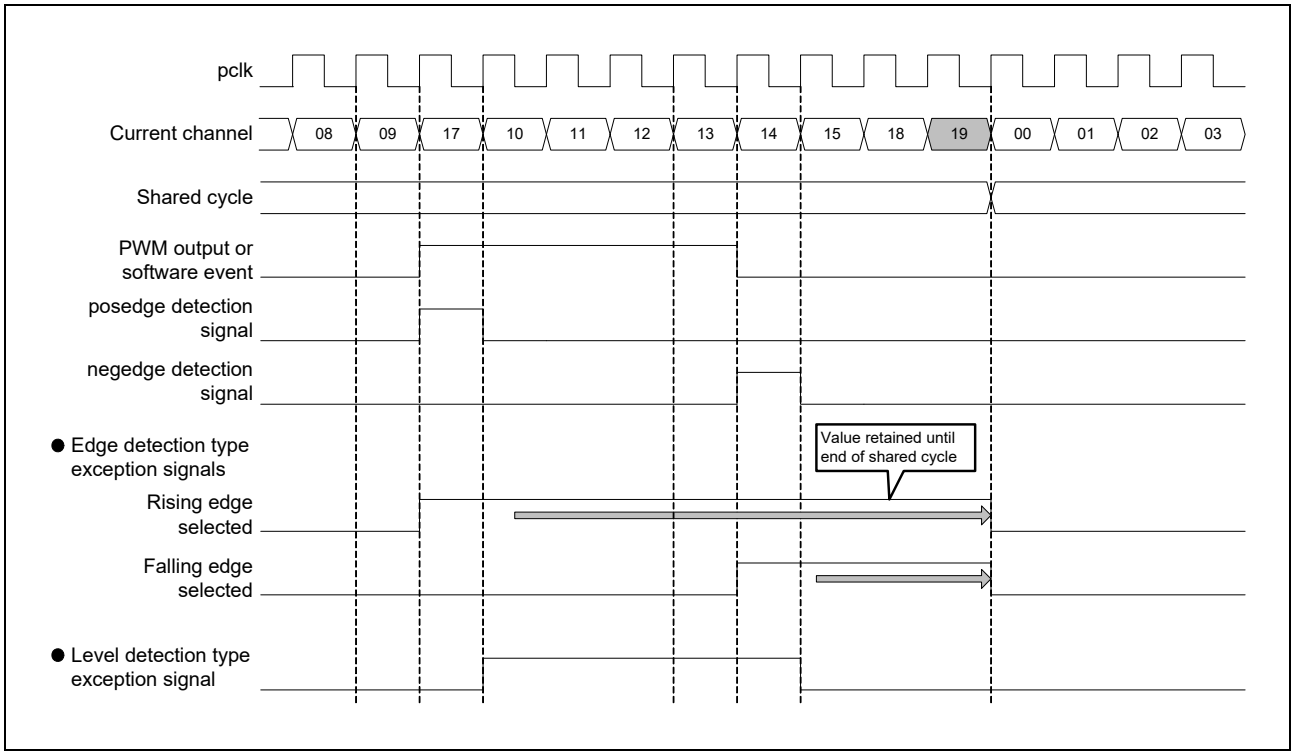


Figure 22.32 Timing Chart of Exception Signals when PWM Output or Software Event is Selected

(1) Stop Exception

The Stop exception processing causes the operations shown in **Table 22.59**.

Table 22.59 Stop Exception Operations

Item	Description
Operation	• Stop exception stops the matching comparator operation. • Stop exception is valid while the matching comparator is operating (valid in states other than disable, dis_wstart, and pre_dis).
Signal change upon exception acceptance	The following signals change simultaneously with state transition (shared cycle boundary). • Match control registers: Initialized (00_B: Match not detected) • PWM output: Inactive level PWM output changes one pclk cycle after state transition because of being F/F output. • Interrupt output: Not changed (low level) • slotID: Initialized (00_H)
Assignment	• Selected from the 32-bit event buses. Refer to **Section 22.5.3.2, Operation: Selection of Event Inputs.** • Separately defined for each pulse generation channel. APAACHNIDSTP[4:0] register is used for setting.
Detection type	• Edge detection (rising edge, falling edge, or both edges) • Separately defined for each pulse generation channel. APAACHNEGSTP[1:0] register is used for setting.
Option	• None
Note	• When Start exception option (APAACHNOPSTR) is valid, a Stop exception does not stop the operation. (A Stop exception stops the operation once, but the operation restarts from the first element.) In this case, the operation can be stopped by setting the APAA0CHEN register to disable the operation of the applicable channel.

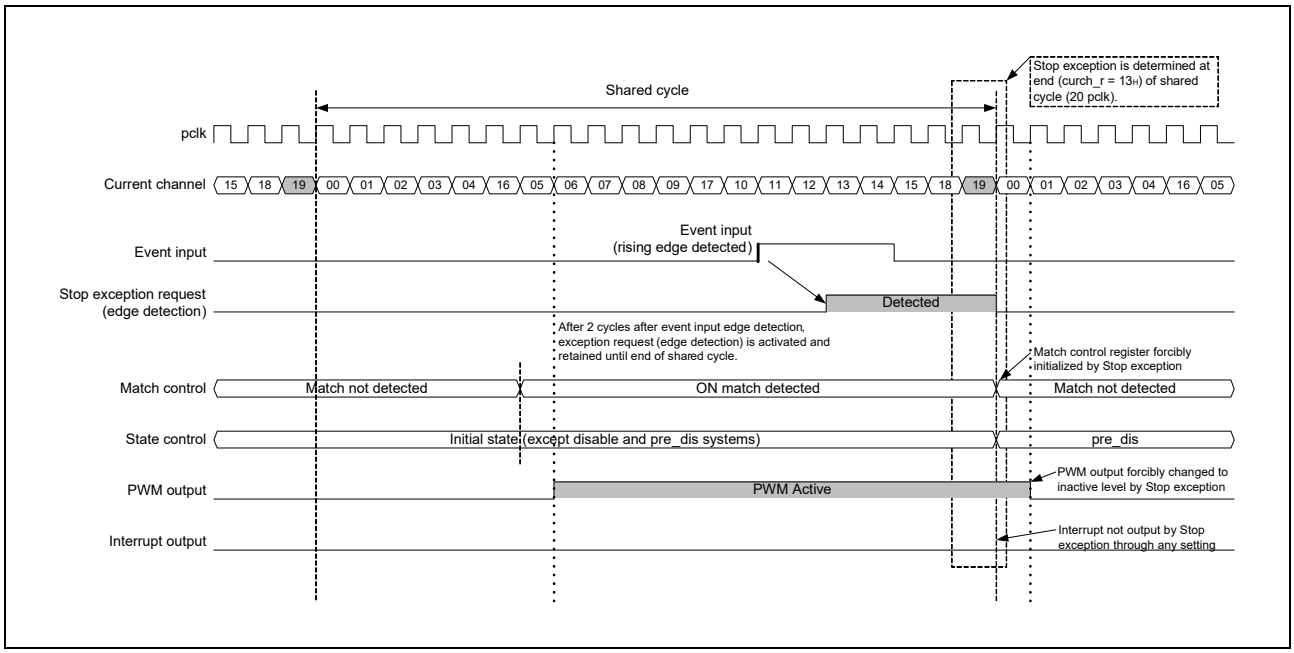


Figure 22.33 Stop Exception Acceptance Timing

(2) Start Exception

The Start exception processing causes the operations shown in **Table 22.60**.

Table 22.60 Start Exception Operations

Item	Description
Operation	• Start exception starts the matching comparator operation. • Start exception is valid only while the matching comparator is stopped and channel operation is enabled (ON) (valid in dis_wstart state).
Signal change upon exception acceptance	No signals change. • Match control registers: Not changed • PWM output: Not changed • Interrupt output: Not changed • slotID: Not changed
Assignment	• Selected from the 32-bit event buses. Refer to **Section 22.5.3.2, Operation: Selection of Event Inputs**. • Separately defined for each pulse generation channel. APAACHNIDSTR[4:0] register is used for setting.
Detection type	• Edge detection (rising edge, falling edge, or both edges) • Separately defined for each pulse generation channel. APAACHNEGSTR[1:0] register is used for setting.
Option	• The Start option (APAACHNOPSTR) is used. – 0: Operation starts triggered by a Start exception after operation is enabled (default) – 1: Operation starts immediately after operation is enabled. Start exception is not required.
Note	• When Start exception option (APAACHNOPSTR) is valid, a Stop exception does not stop the operation. (A Stop exception stops the operation once, but the operation restarts from the first element.) In this case, the operation can be stopped by setting the APAA0CHEN register to disable the operation of the applicable channel.

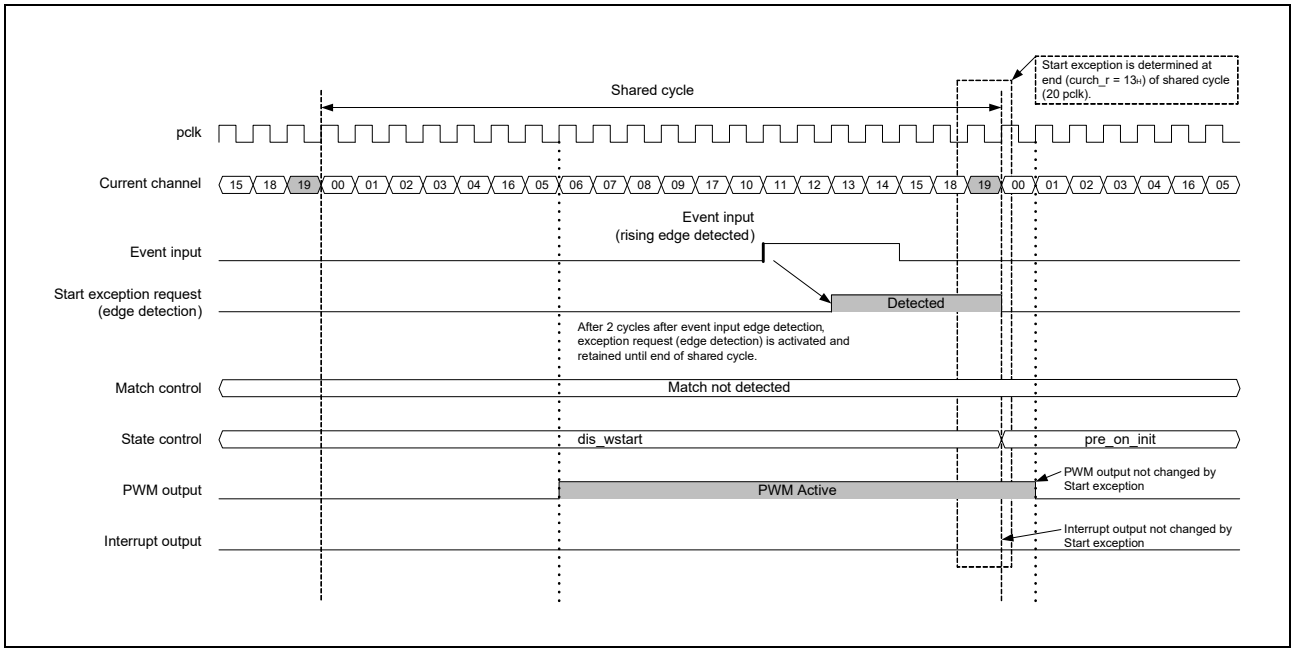


Figure 22.34 Start Exception Acceptance Timing

(3) Restart Exception

The Restart exception processing causes the operations shown in **Table 22.61**.

Table 22.61 Restart Exception Operations

Item	Description
Operation	• Restart exception forcibly initiates transition to the initial element while the matching comparator is operating. • Restart exception is valid only while the matching comparator is not stopped (valid in states other than disable and pre_dis). • When Restart exception is accepted, PWM output is forcibly made inactive simultaneously with state transition. Here, no interrupt is output.
Signal change upon exception acceptance	The following signals change simultaneously with state transition (shared cycle boundary). • Match control registers: Initialized (00_B: Match not detected) • PWM output: Inactive level PWM output changes one pclk cycle after state transition because of being F/F output. • Interrupt output: Not changed (low level) • slotID: Initialized (00_H)
Assignment	• Selected from the 32-bit event buses. Refer to **Section 22.5.3.2, Operation: Selection of Event Inputs**. • Separately defined each pulse generation channel. APAACHNIDRES[4:0] register is used for setting.
Detection type	• Edge detection (rising edge, falling edge, or both edges) • Separately defined for each pulse generation channel APAACHNEGRES[1:0] register is used for setting.
Option	• None
Note	• None

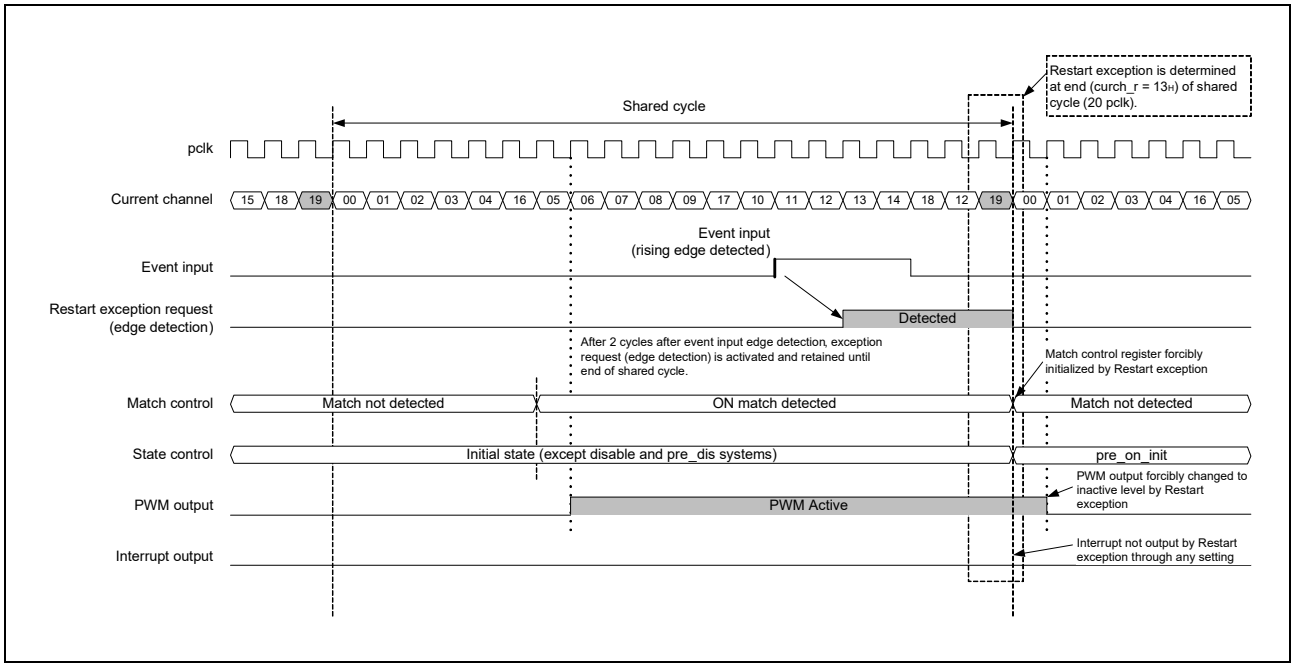


Figure 22.35 Restart Exception Acceptance Timing

(4) Transfer Exception

The Transfer exception processing causes the operations shown in **Table 22.62**.

Table 22.62 Transfer Exception Operations

Item	Description
Operation	• Transfer exception enables state transition upon OFF match. • Transfer exception is valid while ON matching comparison (wait_on system states), OFF matching comparison (wait_off system states), or update to the current element (pre_on_cur and pre_off states) is in progress. • When Transfer exception is accepted with OFF match not yet generated, The Transfer input status flag (APAATRNSTAT register) is set to 1. The value of the APAATRNSTAT register is retained until transition to next elements is initiated and is cleared upon transition to the next elements. • When Transfer exception is accepted with OFF match generated, transition to next elements is initiated. Here, the APAATRNSTAT register is not set.
Signal change upon exception acceptance	The following signals change simultaneously with state transition (shared cycle boundary). • Match control registers: Not changed • PWM output: Not changed • Interrupt output: Not changed (low level) • slotID: Not changed • Transfer input status flag: 0 → 1 (with OFF match not yet generated), or 0 left unchanged (with Transfer exception not yet input before OFF match).
Assignment	• Selected from the 32-bit event buses. Refer to **Section 22.5.3.2, Operation: Selection of Event Inputs**. • Separately defined for each element. APAAELMIDTRN[4:0] register is used for setting.
Detection type	• Edge detection (rising edge, falling edge, or both edges) • Separately defined for each element. APAAELMEGTRN[1:0] register is used for setting.
Option	• The Transfer option (APAAELMOPTRN) is used. – 0: Transfer exception input is required for transition to next elements (default). – 1: Transition to next elements is automatically made upon OFF match (Transfer exception not required).
Note	• Transfer option (APAAELMOPTRN) setting is given priority over element continuation setting (APAAELMCONT); to enable element continuation, clear Transfer option (APAAELMOPTRN) to 0.

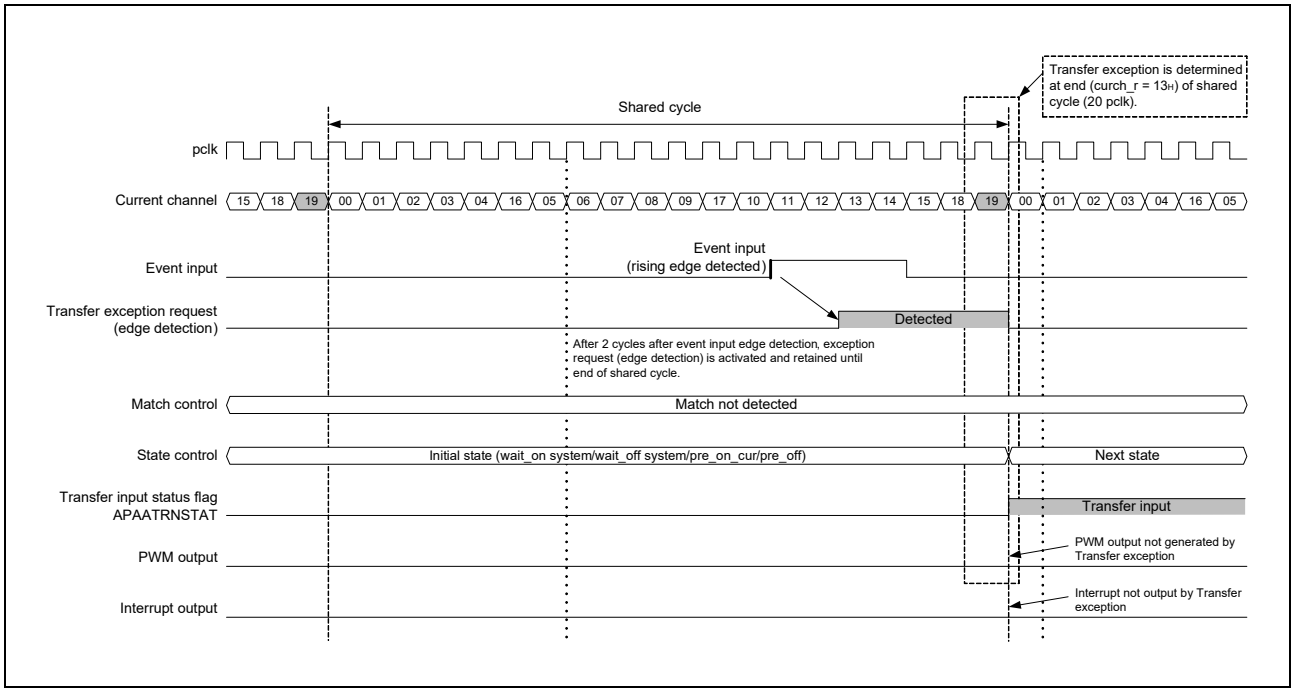


Figure 22.36 Transfer Exception Acceptance Timing

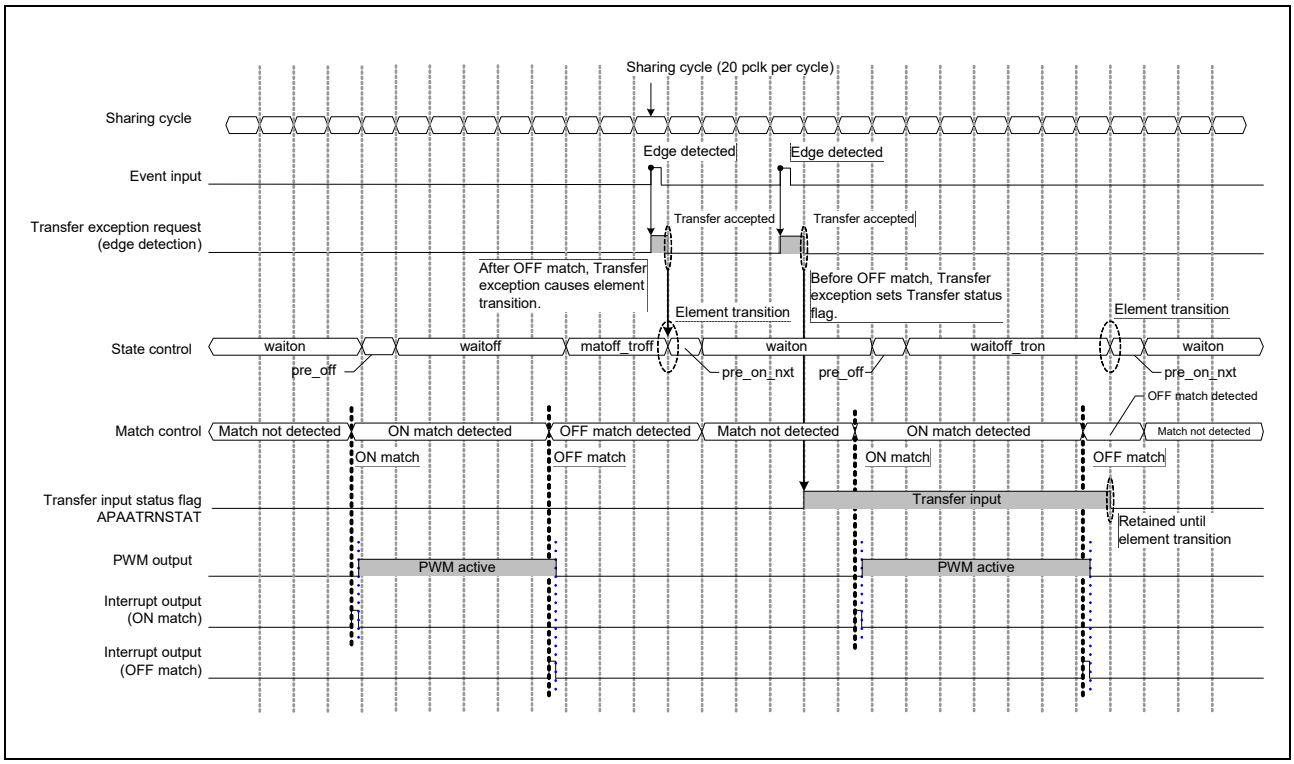


Figure 22.37 Transfer Exception Operation

(5) Force Exception

The Force exception processing causes the operations shown in **Table 22.63**.

Table 22.63 Force Exception Operations (1/2)

Item	Description
Operation	• Force exception forcibly generates a match. • Force exception is valid while ON matching comparison (wait_on system states), OFF matching comparison (wait_off system states), or update to the current element (pre_on_cur and pre_off states) is in progress.
Signal change upon exception acceptance (wait_on system states)	The following signals change simultaneously with state transition (shared cycle boundary). • Case: ON match not yet generated and Force exception (option = default) (refer to **Figure 22.38**) – Match control registers: ON match detected (01_B) – PWM output: Inactive level → active level (not changed in Postpone state) PWM output changes one pclk cycle after state transition because of being F/F output. – Interrupt output: 1-shot active-high pulse is asserted (when ON match interrupt or ON match + OFF match interrupt is set). – slotID: Set to 13_H • Case: ON match not yet generated and Force exception (forced OFF match option) (refer to **Figure 22.39**) – Match control registers: OFF match detected (10_B) – PWM output: Not changed – Interrupt output: 1-shot active-high pulse is asserted (when OFF match interrupt or ON match + OFF match interrupt is set). – slotID: Set to 13_H • Case: ON match generated and Force exception (option = default) (refer to **Figure 22.41**) – Match control registers: Not changed – PWM output: Not changed – Interrupt output: Not changed (low level) – slotID: Set to 13_H • Case: ON match generated and Force exception (forced OFF match option) (refer to **Figure 22.44**) – Match control registers: OFF match detected (10_B) – PWM output: Active level → inactive level (not changed in Postpone state) PWM output changes one pclk cycle after state transition because of being F/F output. – Interrupt output: 1-shot active-high pulse is asserted (when OFF match interrupt or ON match + OFF match interrupt is set). – slotID: Set to 13_H • Case: OFF match generated (Force exception: forced OFF match option) and Force exception (refer to **Figure 22.43**) – Match control registers: Not changed – PWM output: Not changed – Interrupt output: Not changed (low level) – slotID: Set to 13_H

Table 22.63 Force Exception Operations (2/2)

Item	Description
Signal change upon exception acceptance (wait_off system states)	The following signals change simultaneously with state transition (shared cycle boundary). • Case: OFF match not yet generated and Force exception (not depending on option setting) (refer to **Figure 22.40**) – Match control registers: OFF match detected (10_B) – PWM output: Active level → Inactive level (not changed in Postpone state) PWM output changes one pclk cycle after state transition because of being F/F output. – Interrupt output: 1-shot active-high pulse is asserted (when OFF match interrupt or ON match + OFF match interrupt is set). – slotID: Set to 13_H • Case:OFF match generated and Force exception (not depending on option setting) (refer to **Figure 22.42**) – Match control registers: OFF match detected (10_B) – PWM output: Not changed – Interrupt output: Not changed (low level) – slotID: Set to 13_H
Assignment	• Selected from the 32-bit event buses. Refer to **Section 22.5.3.2, Operation: Selection of Event Inputs**. • Separately defined for each element. APAAELMIDFOR[4:0] register is used for setting.
Detection type	• Edge detection (rising edge, falling edge, or both edges) • Separately defined for each element. APAAELMEGFOR[1:0] register is used for setting.
Option	• The Force option (APAAELMOPFOR) is used. – 0: Next match is forcibly generated. ON waiting → ON match is forcibly generated; OFF waiting → OFF match is forcibly generated (default). – 1: OFF match is forcibly generated. ON waiting → OFF match is forcibly generated; OFF waiting → OFF match is forcibly generated.
Note	• None

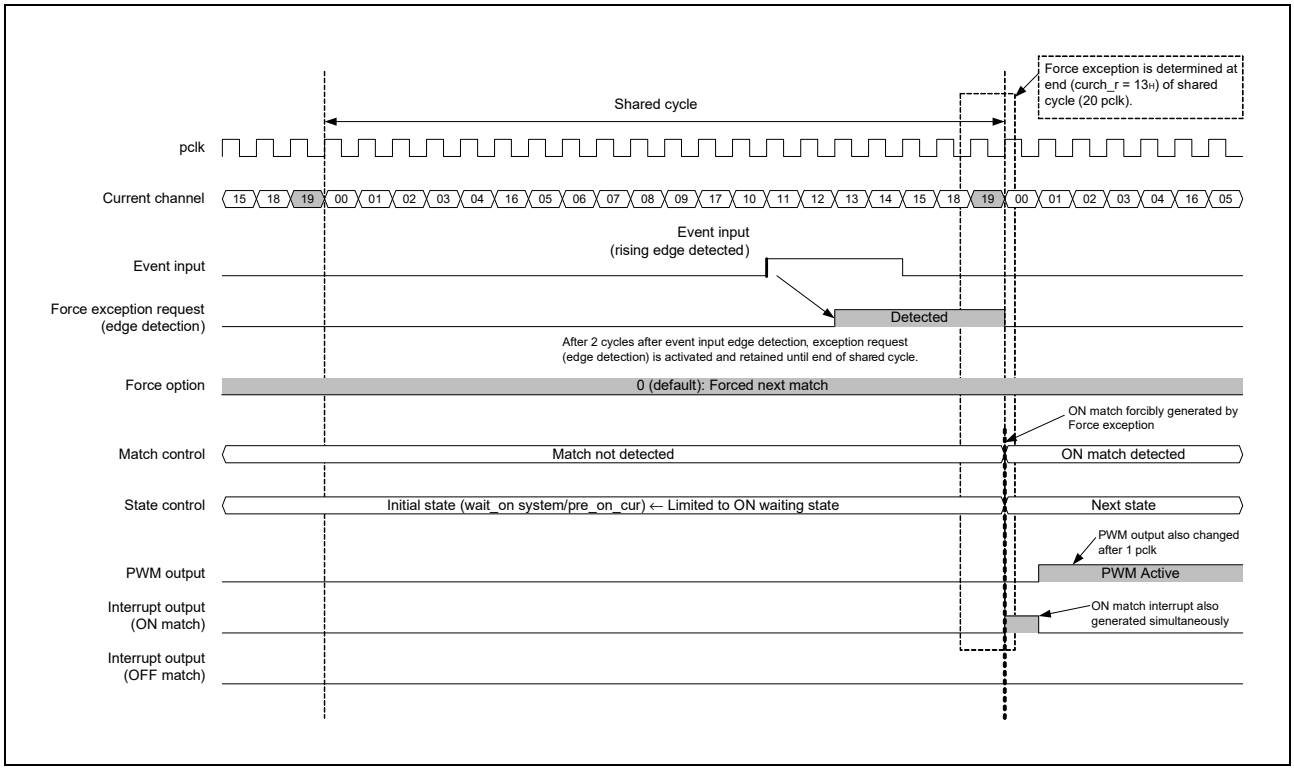


Figure 22.38 Force Exception Operation 1 (Match Not Detected → ON Match Detected)

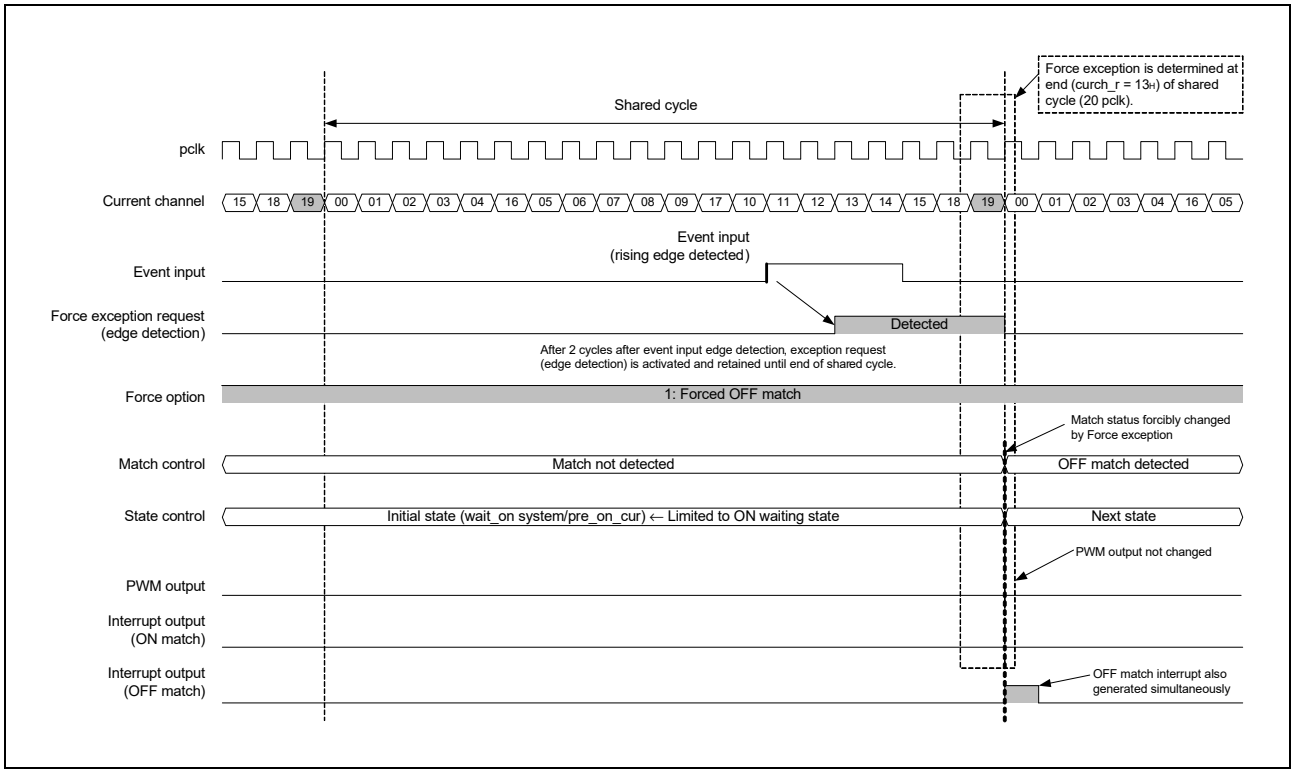


Figure 22.39 Force Exception Operation 2 (Match Not Detected → OFF Match Detected)

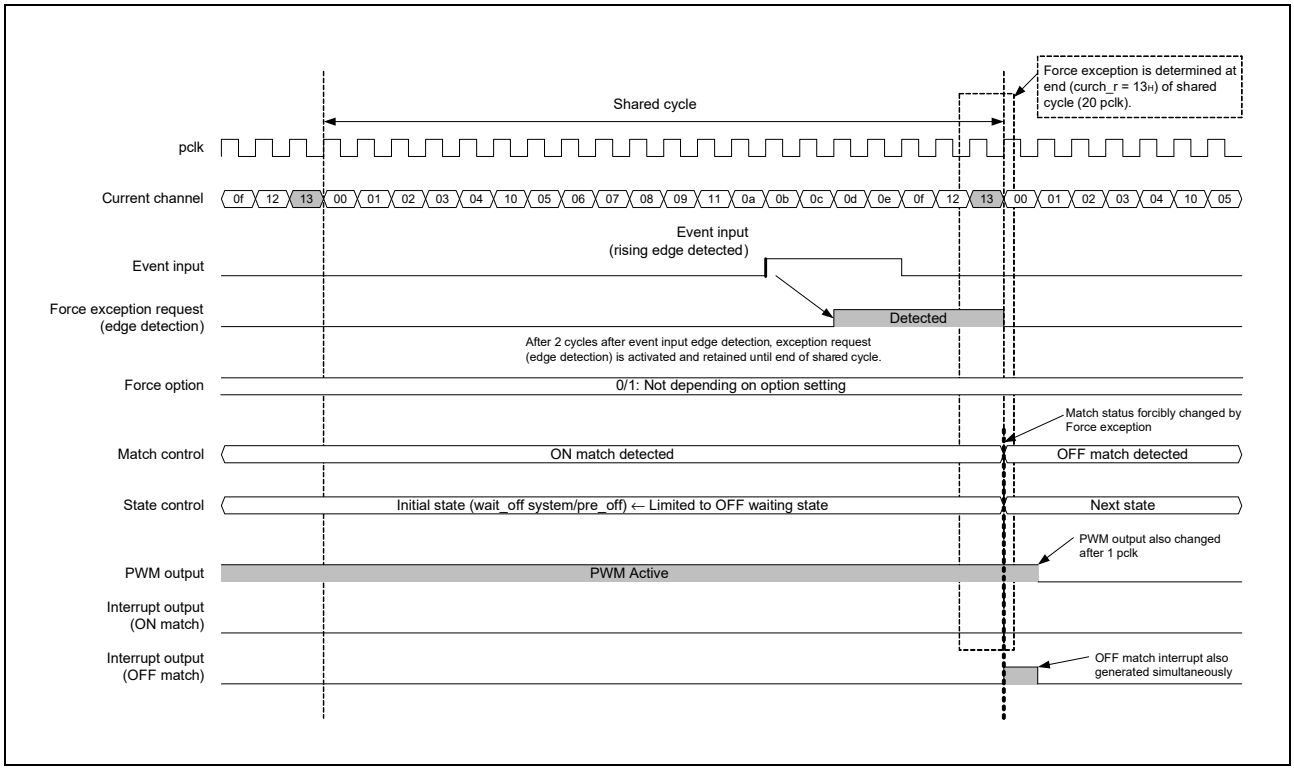


Figure 22.40 Force Exception Operation 3 (ON Match Detected → OFF Match Detected)

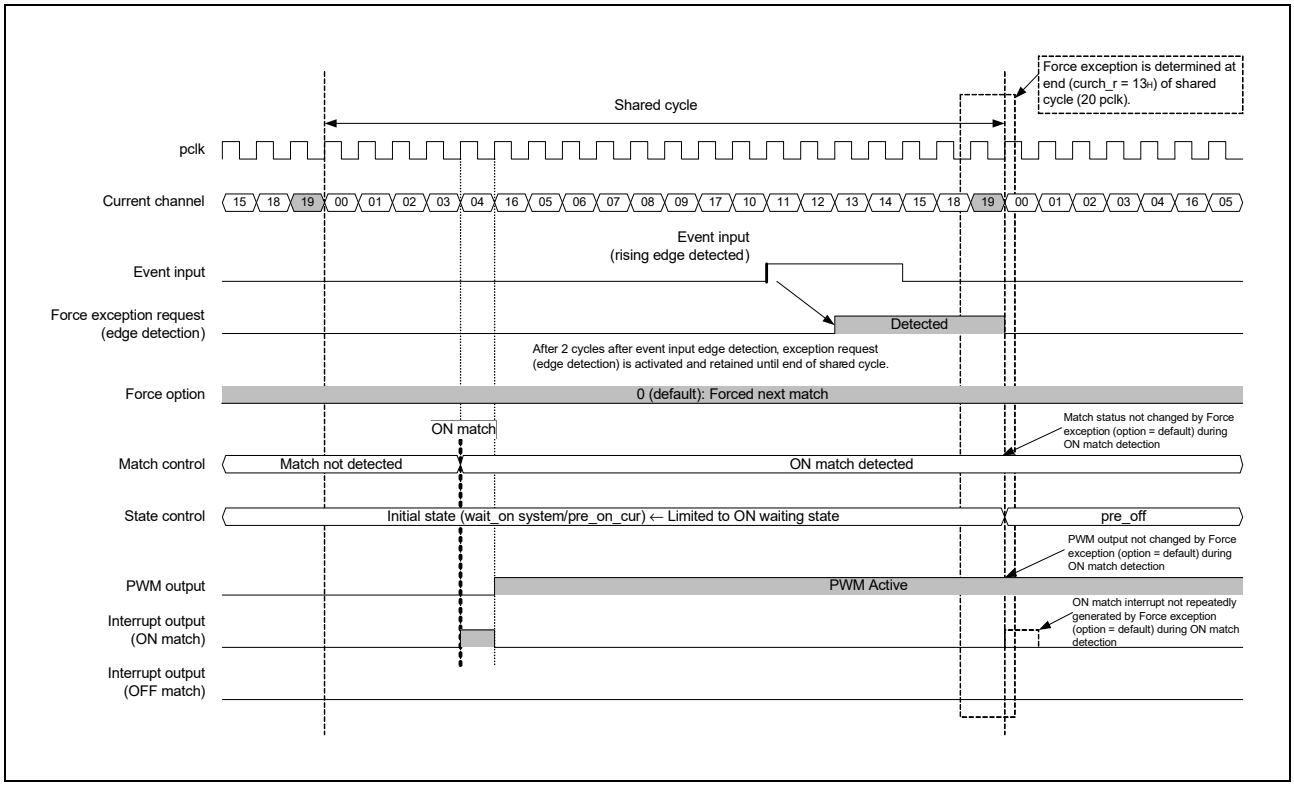


Figure 22.41 Force Exception Operation 4 (Repeated ON Match Detection)

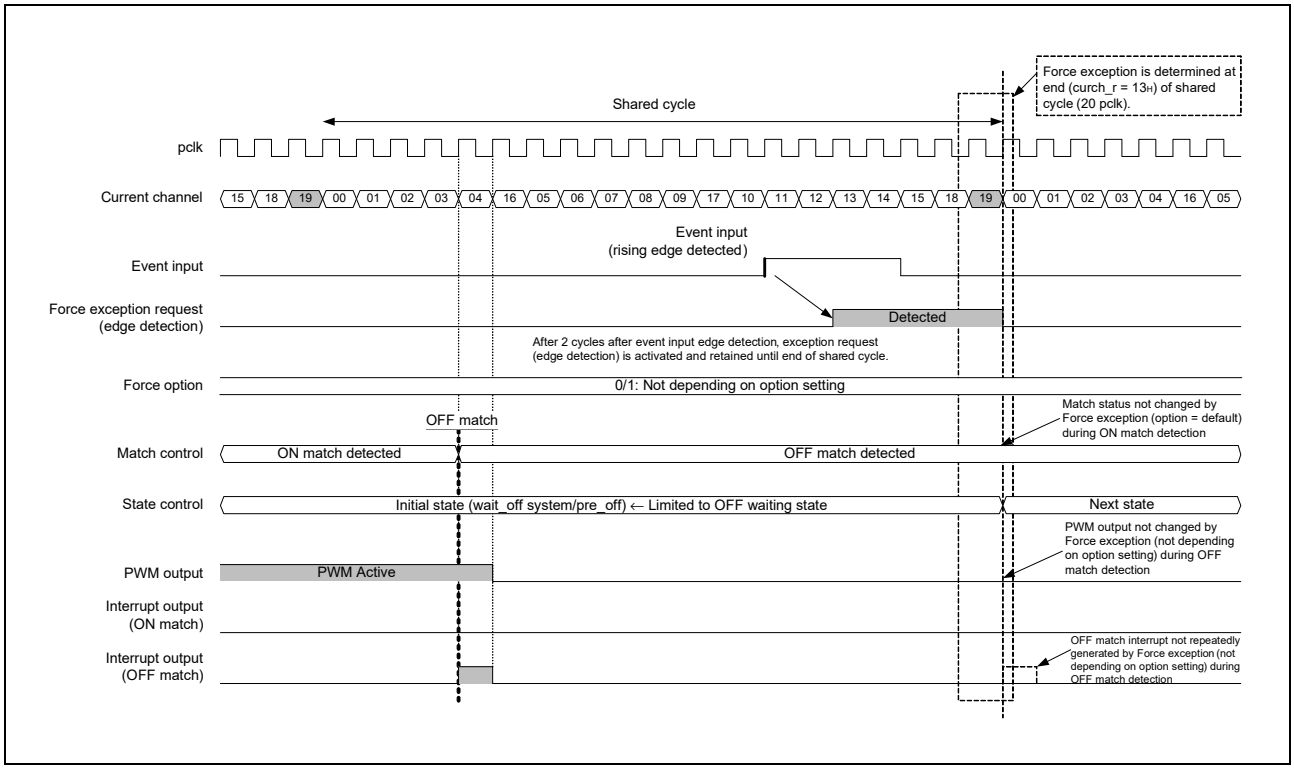


Figure 22.42 Force Exception Operation 5 (Repeated OFF Match Detection 1)

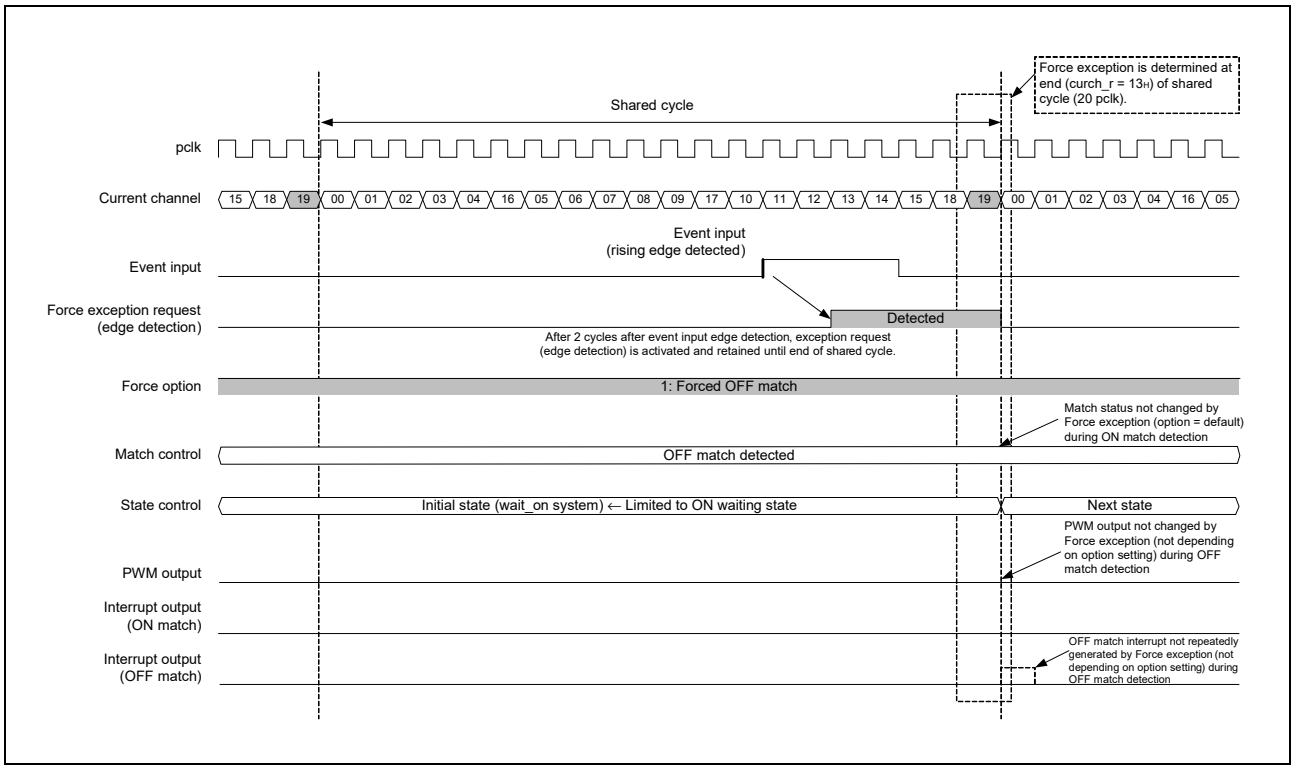


Figure 22.43 Force Exception Operation 6 (Repeated OFF Match Detection 2)

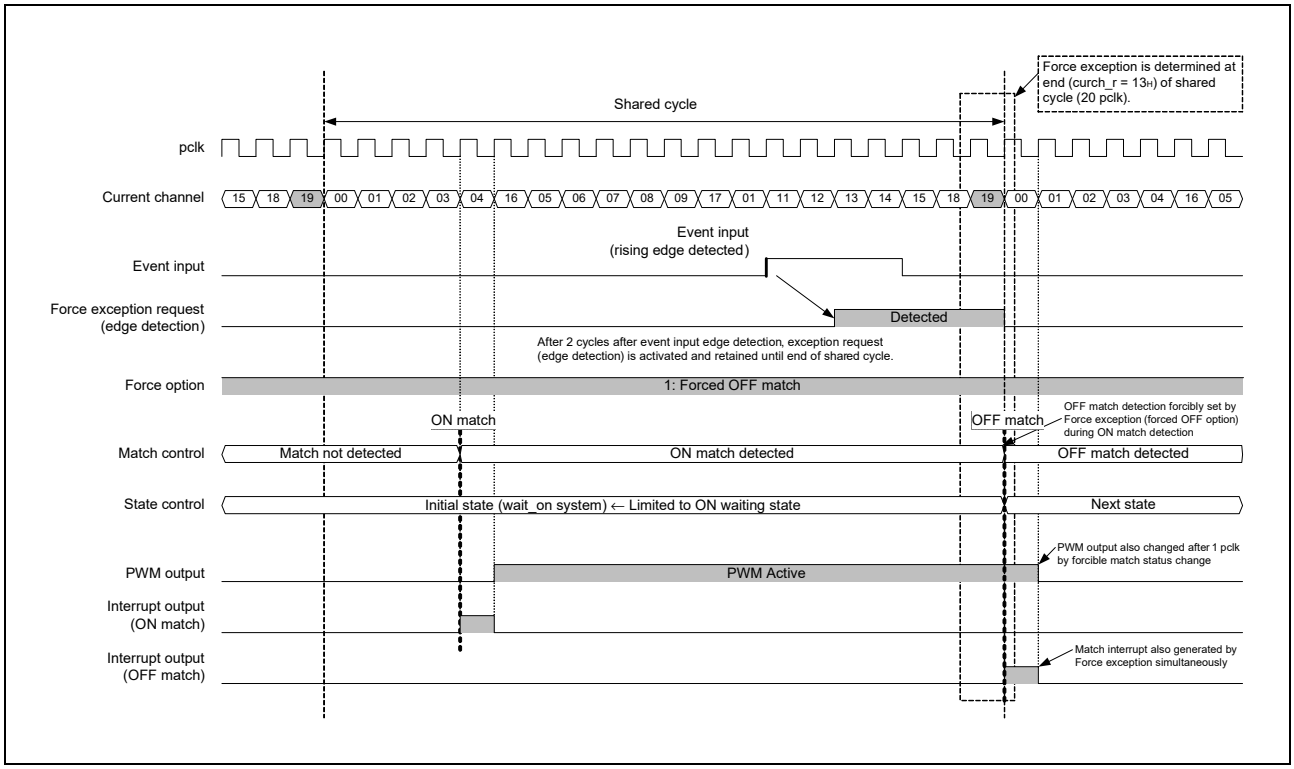


Figure 22.44 Force Exception Operation 7 (Conflict Between ON Match Detection and OFF Match Detection)

(6) Postpone Exception

The Postpone exception processing causes the operations shown in **Table 22.64**.

Table 22.64 Postpone Exception Operations

Item	Description
Operation	• Postpone exception temporarily postpones the operation that should normally takes place upon a match. • Postpone exception is valid while ON matching comparison (wait_on system states) or OFF matching comparison (wait_off system states) is in progress. • When Postpone exception turns active and is accepted as an exception, the Postpone status flag (APAAPPNSTAT register) is set to 1 simultaneously with state transition and the status is retained while the matching comparator is in the Postpone state. When Postpone exception turns inactive, the Postpone state is canceled simultaneously with state transition and the APAAPPNSTAT register is cleared to 0. • While the matching comparator is in the Postpone state, PWM output does not change. However, ON matching comparison or OFF matching comparison continues and PWM output begins to change according to the match status from the next state after Postpone state is canceled. • While Postpone exception is accepted, ON match interrupt or OFF match interrupt is output according to ON/OFF match detection or Force exception input.
Signal change upon exception acceptance	The following signals change simultaneously with state transition (shared cycle boundary). • Match control registers: Not changed • PWM output: Not changed • Interrupt output: Not changed (low level) • slotID: Not changed • Postpone status flag: 0 → 1
Signal change in Postpone state	The following signals change simultaneously with state transition (shared cycle boundary). • Match control registers: Changed according to match/Force • PWM output: Not changed • Interrupt output: Changed according to match/Force • slotID: Changed according to match/Force • Postpone status flag: Not changed (remains 1)
Signal change upon Postpone cancelation	The following signals change simultaneously with state transition (shared cycle boundary). • Match control registers: Changed according to match/Force • PWM output: Changed according to match/Force • Interrupt output: Changed according to match/Force • slotID: Changed according to match/Force • Postpone status flag: 1 → 0
Assignment	• Selected from the 32-bit event buses. Refer to **Section 22.5.3.2, Operation: Selection of Event Inputs**. • Separately defined for each element. APAAELMIDPPN[4:0] register is used for setting.
Detection type	• Level detection (either high or low) • Separately defined for each element. APAAELMLVPPN register is used for setting.
Option	• None
Note	• Since Postpone exception is level-sensitive, the assigned signal (one of event inputs or PWM outputs) needs to be high or low for 1 or more shared cycles (pclk*20). If the input level cannot be retained for the specified duration, the input may fail to be accepted as Postpone exception input.

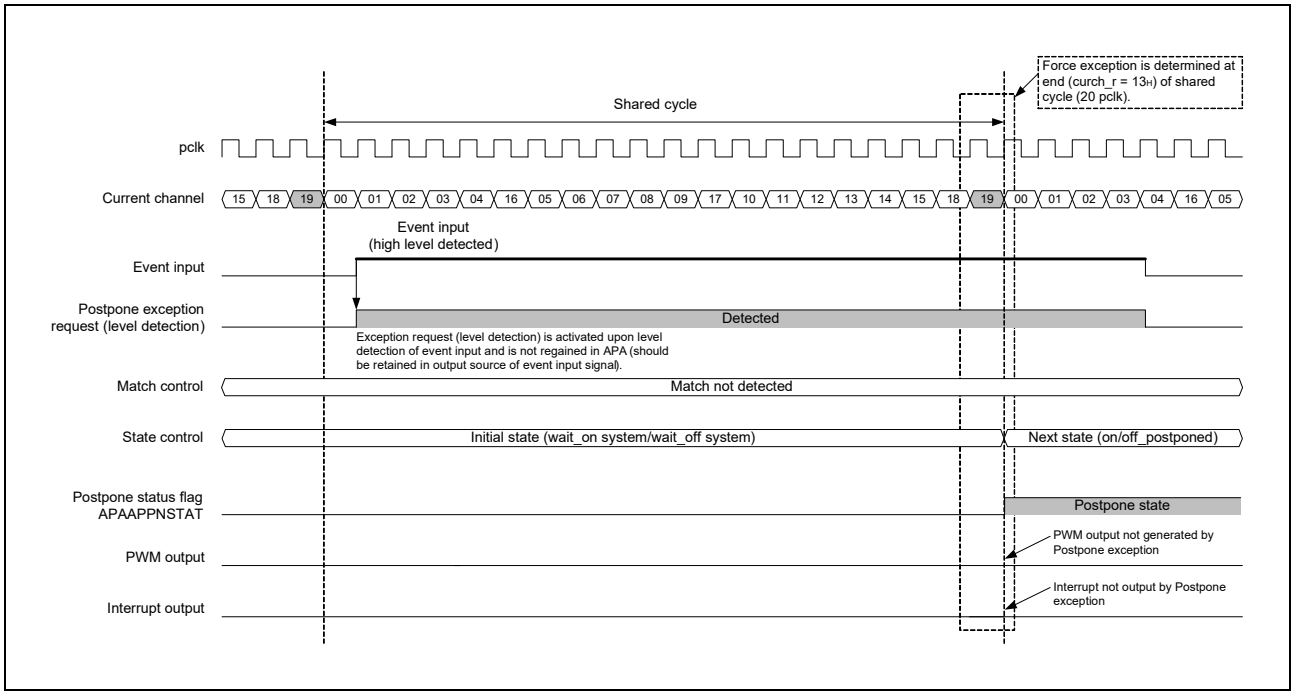


Figure 22.45 Postpone Exception Acceptance Timing

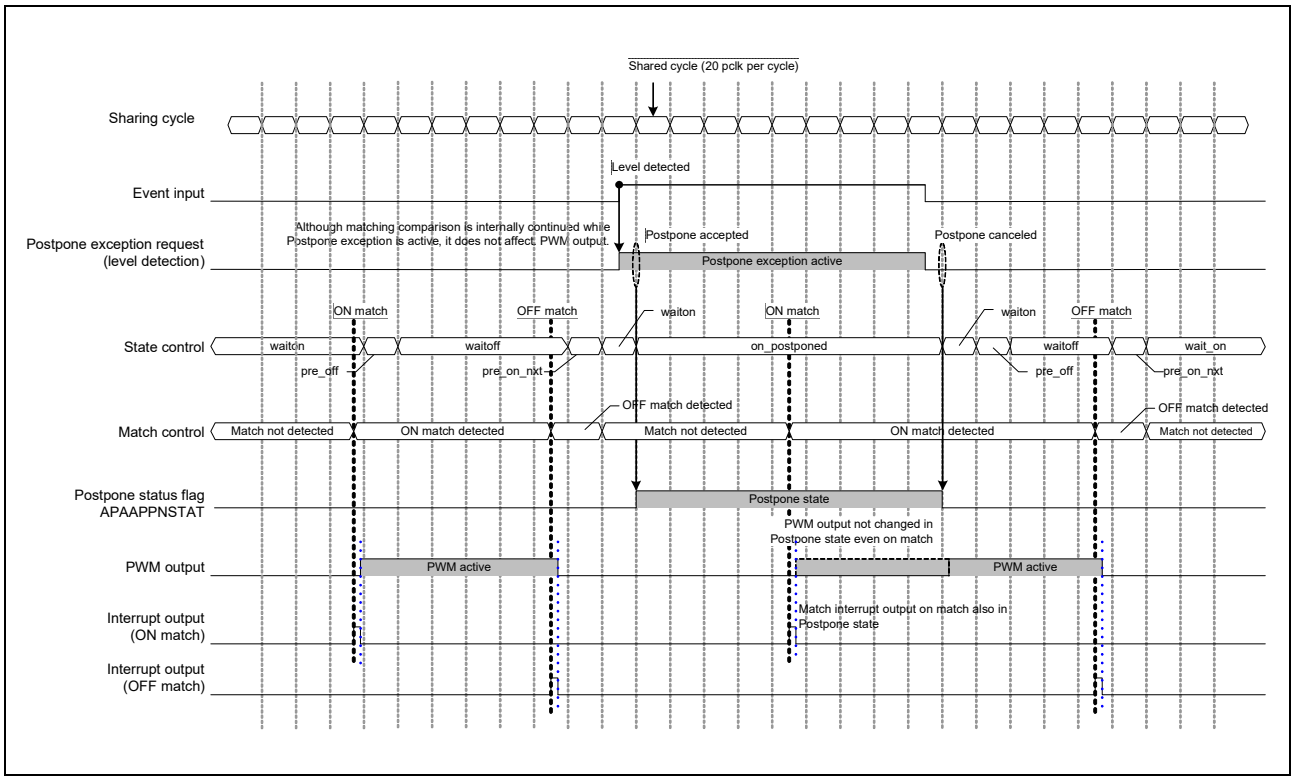


Figure 22.46 Postpone Exception Operation

22.5.4.4 Operation: Conflict between Matching Comparator and Event Decoder

When multiple input conditions shown in **Table 22.64** conflict in the same sharing cycle, the input conditions are handled as follows.

(1) Conflicts involving Channel OFF, Stop Exception, and Restart Exception

Channel OFF, Stop exception, and Restart exception are accepted according to the following priority order.

- Channel OFF > Stop exception > Restart exception

When Channel OFF, Stop exception, or Restart exception conflicts with the other input conditions, only one of the Channel OFF, Stop exception, and Restart exception is accepted according to the above priority order and the other conditions are ignored.

(2) Conflicts involving Start Exception

When Start exception, which is normally accepted only in the dis_wstart state, conflicts with Restart exception, both exceptions are accepted (in this state, Restart exception and Start exception behave in the same way).

In the other states, Start exception is ignored.

(3) Conflicts involving Transfer Exception

When Transfer exception has not yet been accepted in any state in which Transfer exception can be accepted, Transfer exception is accepted independently of the other input conditions (except for Channel OFF, Stop exception, and Restart exception), which affects the pertinent state transition. Here, the other input conditions can also be accepted. Contrarily, when Transfer exception has been accepted in any state in which Transfer exception can be accepted and another Transfer exception is input, it is ignored.

(4) Conflicts involving Match Detection and Force Exception

When Force exception is input in the same sharing cycle in which match is detected, Force exception is ignored.

However, when ON match is detected in the ON waiting state and Force exception (forced OFF option) is input, Force exception (forced OFF option) is accepted (refer to **Figure 22.42**).

When Force exception is input with match not yet detected, Force exception is accepted.

(5) Conflicts involving Postpone Exception (Postpone canceled state → Postpone state)

Postpone exception is accepted when match has not yet been detected and Postpone exception does not conflict with Force exception.

Postpone exception is ignored in the pertinent state when match has not yet been detected or when Force exception is input in conflict with Postpone exception.

(6) Conflicts involving Postpone Exception (Postpone state → Postpone canceled state)

Even when Postpone exception conflicts with match or Force exception, associated operations are completely executed.

22.5.5 Element control

22.5.5.1 Features and Overview of functions

This function generates control signals for the element RAM and the calculation device. In order to allow the elements to be shared by all pulse control channels, the following processing and control are made:

- All element information can be read and written through the P-Bus. Element data can be read and written whenever pulse generation channels are not generating pulses. If pulse generation is in progress, element data can be read and written only during the time when pulse generation channels are not accessing the element RAM. This time period is controlled by the event time division signals generated by the event bus.
- Each pulse generation channel can only read the elements. When an ON match, an OFF match, or exception processing (Transfer, Restart, Force, or Start) is generated in a pulse generation channel, the channel issues an element update request (a read request) to the element RAM. Element update requests align with the time division governed by the event time division signal, so that they are controlled not to conflict with reads and writes through the P-Bus and other pulse generation channels.
- If the next element has a match condition of relative designation, the designated value cannot be compared with the reference input as it is; it needs to be converted to a comparison baseline (absolute value). The element decoder is equipped with the calculation device (addition device) for this conversion. As with the element RAM, this calculation device is shared by different pulse generation channels.

Figure 22.47 shows the configuration of the element control circuit.

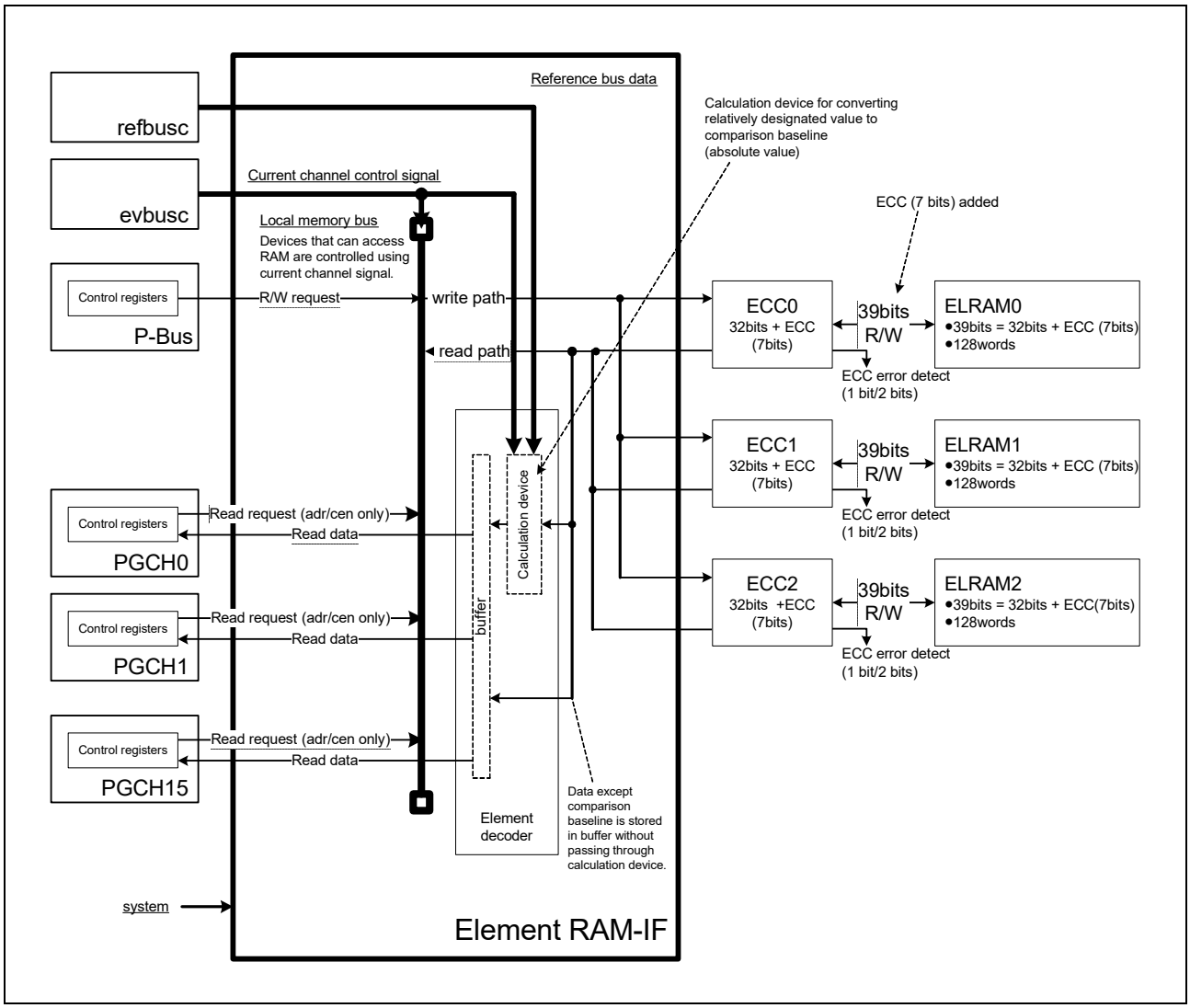


Figure 22.47 Configuration of Element Control Circuit

22.5.5.2 Operation

Access to the element RAM (from the P-Bus or pulse generation channels 0 to 15) is performed using time division as shown in **Figure 22.48**. Access from the P-Bus is held in the wait state until access to the element RAM is enabled.

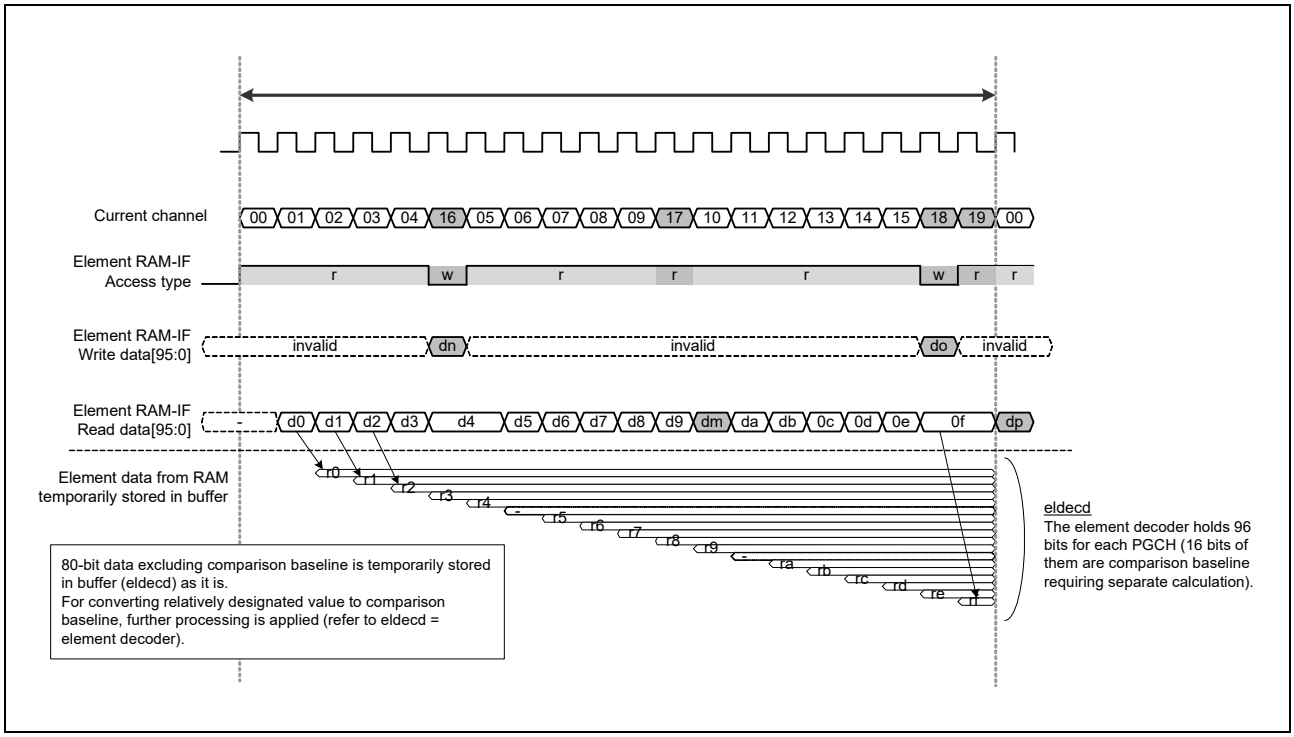


Figure 22.48 Access to Element RAM

22.5.5.3 Changing Starting Element Number during Operation

When the starting element number setting bits APAA0CCGAn.APAACHNFSTEL[6:0] (n = 0 to 15) are modified while a pulse generation channel is operating, the newly set number is applied when the starting element number is referenced to next (pre_on_init state). Therefore, to change the starting element when the starting element number is referenced to next, modify the APAA0CCGAn.APAACHNFSTEL[6:0] (n = 0 to 15) bits before transition to pre_on_init state (refer to **Figure 22.50**). For details of state transition, refer to **(2) State Control** in **Section 22.5.4.2, Operation: Matching Comparator**.

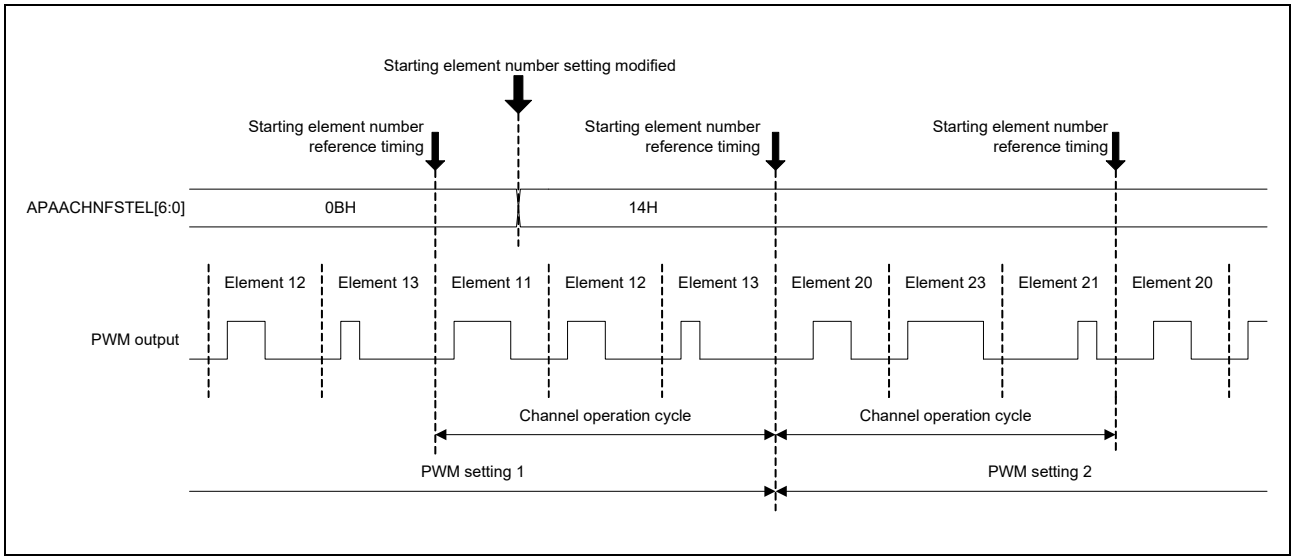


Figure 22.49 Operation Example when Starting Element is Changed during Operation

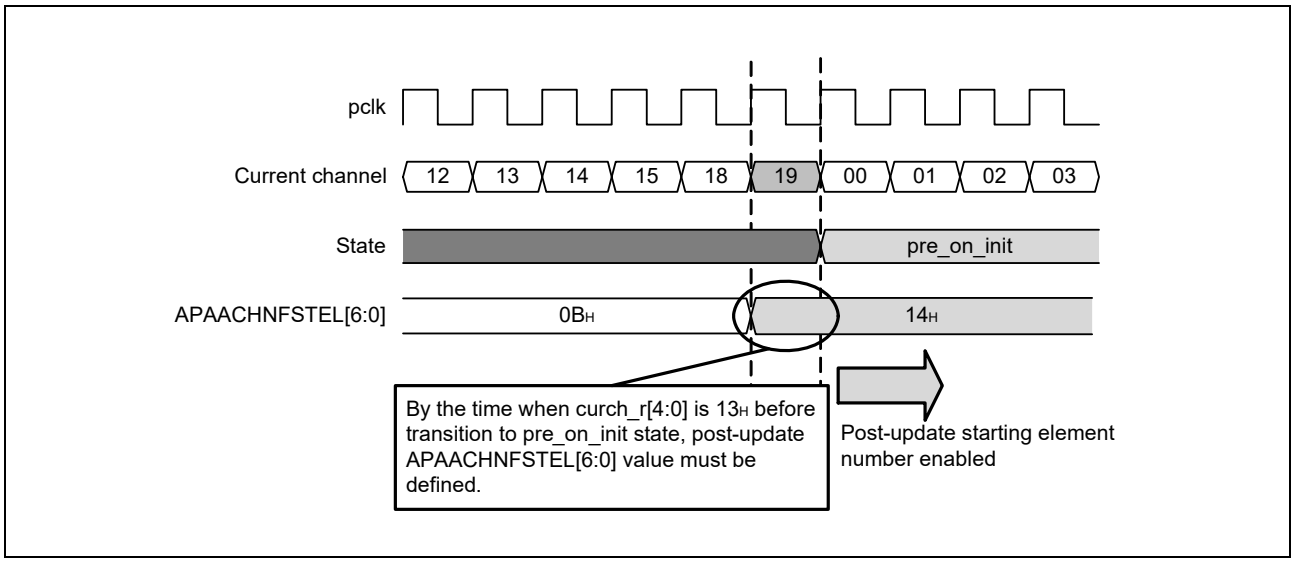


Figure 22.50 Timing of Changing Starting Element Number

Figure 22.51 shows the procedure for modifying both the starting element number and unused element RAM during operation. If this procedure is not followed, pulse generation channels might behave in an unintended way.

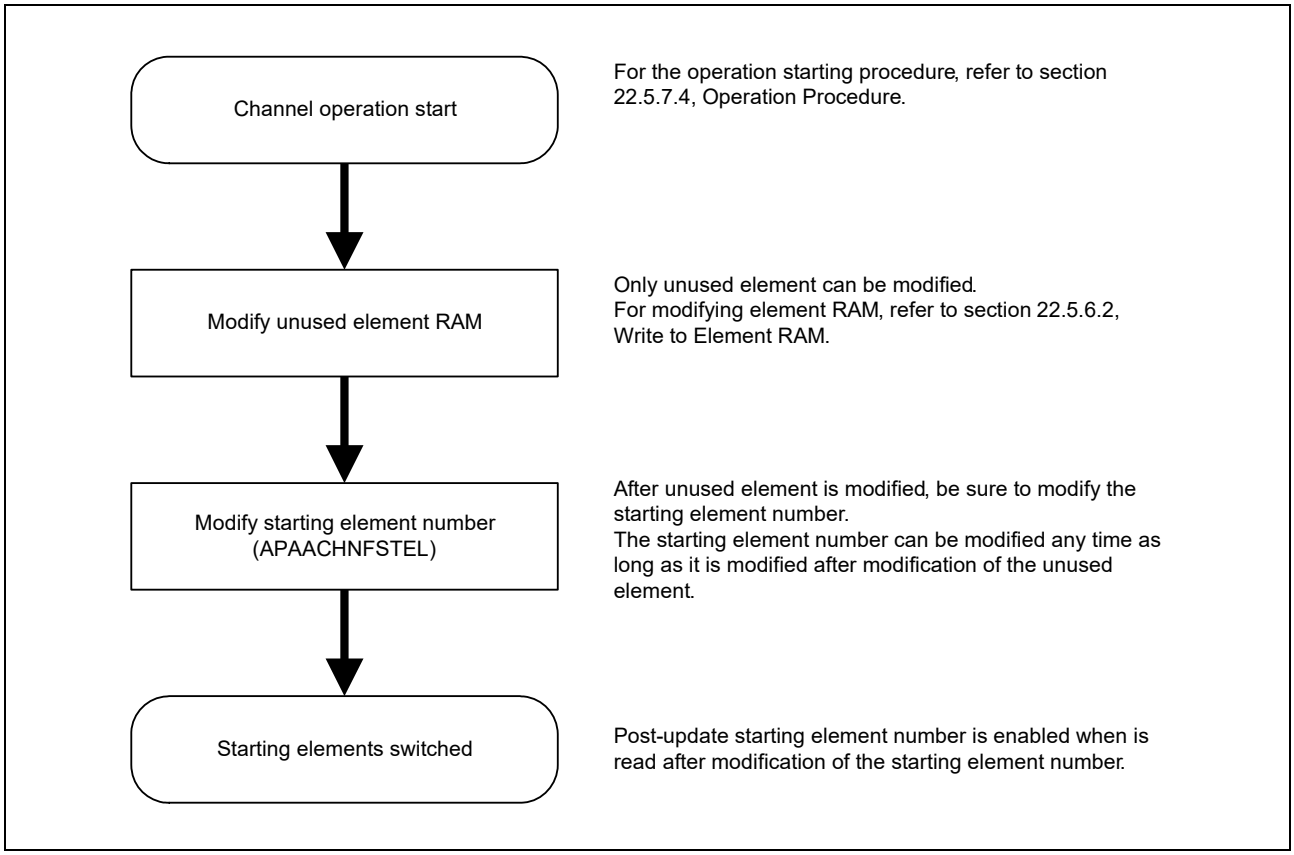


Figure 22.51 Procedure for Changing Starting Element and Unused Element during Operation

22.5.5.4 Operation: Element Decoder

In element update processing triggered by exception processing or a match occurrence, the match condition in the element that is read from the element RAM can be a value of relative designation. By definition, the value of relative designation cannot be compared with the reference input as it is; it must be converted to a comparison baseline (absolute value). The element decoder allows the calculation device used for conversion to be shared by multiple pulse generation channels. It operates as follows:

- Absolutely designated values are not converted.
- In the case of relative comparison, calculation is made based on the specified reference value when a match detection starts.
- In the case of relative designation, the reference input must satisfy the following conditions (**Figure 22.52**):
  - Quasi-monotone increasing waveform (has a “saw” shape).
  - The maximum value is fixed.
- If overflow occurs when converting from the relatively designated value to the comparison baseline, the portion exceeding the maximum value becomes the comparison baseline (**Figure 22.53**). The maximum value should be specified in advance for each reference.
- In the case of relative designation, the maximum value setting (APAA0RFMX0, APAA0RFMX1, APAA0RFMXsw registers) must satisfy the following conditions (**Figure 22.53**):
  - The value of relative designation (APAAELMRBDATON[15:0] or APAAELMRBDATOFF[15:0]) must not exceed the maximum value that the reference input can take.
  - The maximum value must be equal to the maximum value that the corresponding reference input can take. If the source counter value of the corresponding reference input is larger than the dynamic range of APA, care must be taken on the relationship with the reference input value of APA at an overflow of the source counter.
- Relative designation is valid only for references 0, 1, and sw. If relative designation is made for references 2 to 9, the designated value is used as the comparison baseline.

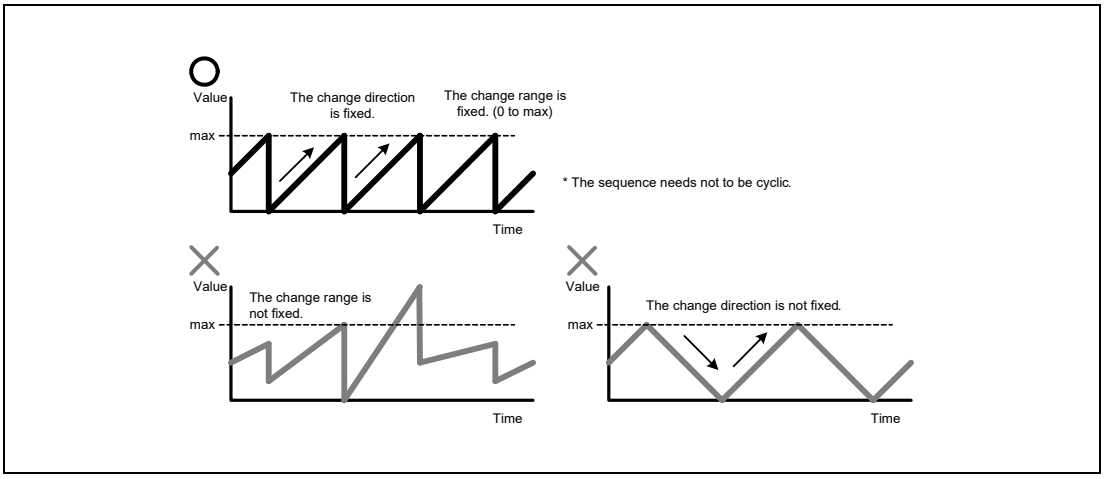


Figure 22.52 Reference Waveform Suitable for Relative Designation

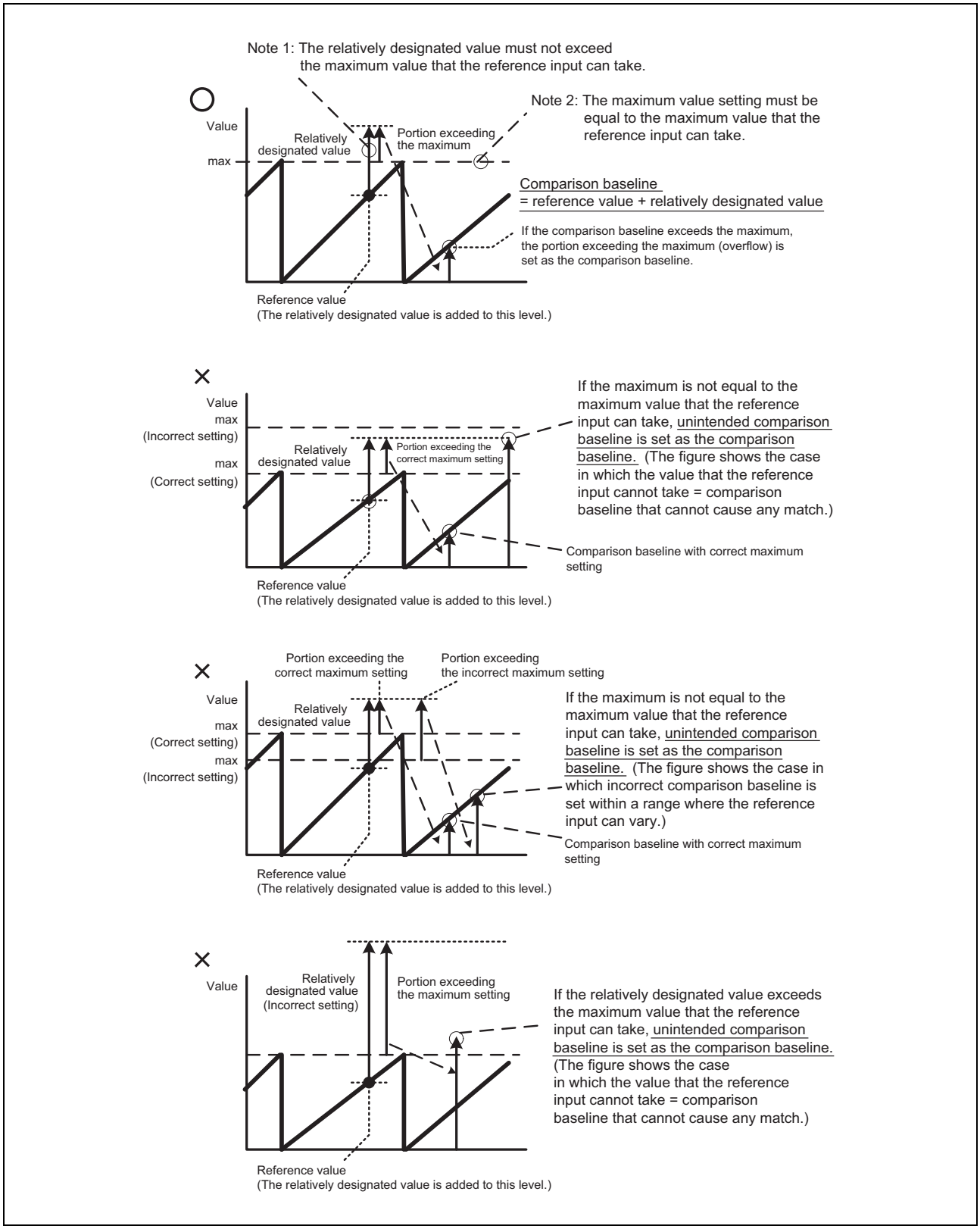


Figure 22.53 Handling of Overflow in Relative Designation

22.5.6 P-Bus-I/F

22.5.6.1 Features and Overview of Functions

The control registers in the APA (refer to **Table 22.8**, **Table 22.14**, **Table 22.20**, **Table 22.27**, and **Table 22.43**) are writable and readable via the P-Bus and the contents of the registers are held in the P-Bus-I/F. The P-Bus-I/F generates the signals to be supplied to sub-modules such as the reference bus and event bus.

22.5.6.2 Write to Element RAM

- Since the element RAM is read also by pulse generation channels, access to RAM is limited to the time in which the element RAM is not used by any pulse generation channel. The element RAM can only be read or written to in the event slots that are not assigned to pulse generation channels according to event time division timing.
- In writing to the element RAM, write 3 words of data continuously (32 low-, 32 middle-, and 32 high-order bits, in this order) to store up element data in the registers, which completes preparation for write access to the element RAM. If the words are written in the wrong order, or the element number is changed or the element RAM is read in the middle of write, unintended data is written.
- After the write preparation is completed, a wait time is produced until an event slot becomes available in which the P-Bus has priority. A wait time length depends on the APAA0EN register setting: 3 PCLK cycles when APAACMENAPA is 0 and 3 to 9 PCLK cycles when APAACMENAPA = 1.
- As soon as an event slot becomes available, data is written to the element RAM (1 cycle).
- Writing data to the registers during operation is possible only for the element not currently used. If data is written for the currently used element, the pulse generation channel in operation might behave in an unintended way.

22.5.6.3 Read from Element RAM

- Similar to write, the element RAM is read according to time division timing from the event bus. (Access is possible only in the event slots in which the element RAM is not used by any pulse generation channel.)
- Data can be read from any location in the element RAM.
- The element data (96 bits) having been read from RAM is stored in the buffer (read buffer). The data is retained until read to another element (or write to the same address read) occurs.
- When accessing the address from which data has not yet stored in the buffer, a wait time is produced until an event slot becomes available in which the P-Bus has priority. A wait time length depends on the APAA0EN register setting: 3 PCLK cycles when APAACMENAPA is 0 and 3 to 9 PCLK cycles when APAACMENAPA = 1.
  When accessing the address from which data has stored in the buffer, no wait time is produced irrespective of the APAACMENAPA setting.

22.5.7 Usage Notes

22.5.7.1 Matching Condition Setting 1

For a condition setting between an ON match and an OFF match, consider the resolution of the reference, and set the matching condition with a pulse width of 2 sharing cycles (pclk * 40) or more.

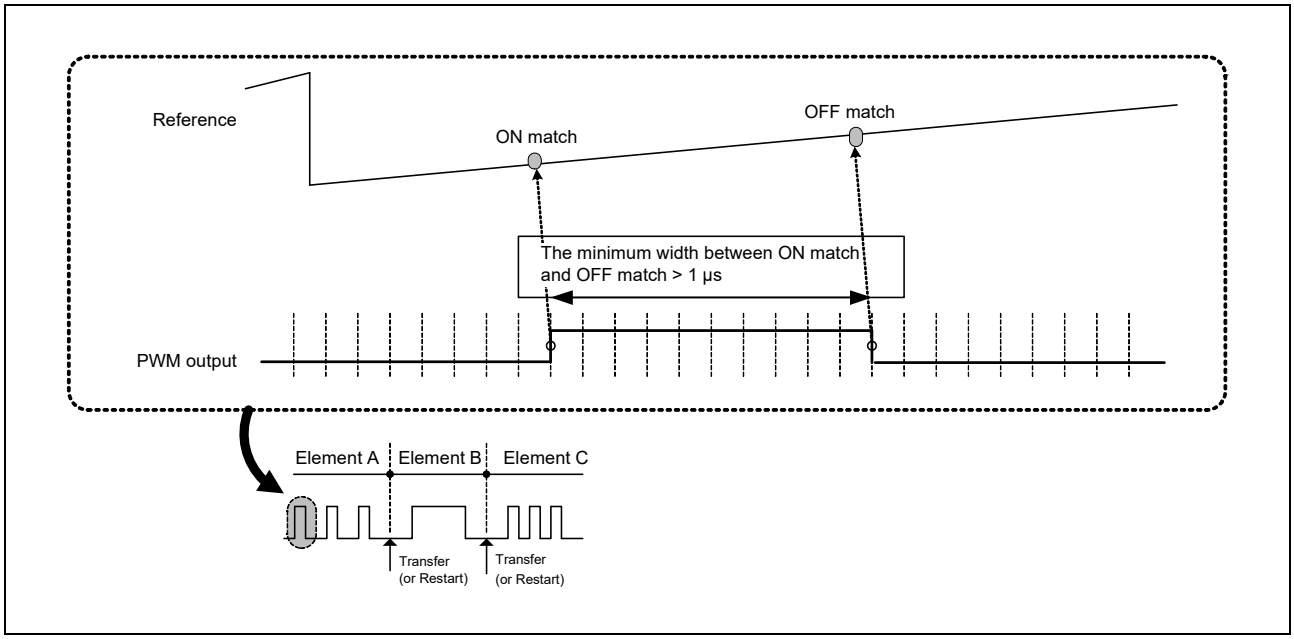


Figure 22.54 Restriction of Matching Condition Setting

22.5.7.2 Matching Condition Setting 2 (ADC Input)

In cases where the reference input takes discrete values (e.g. ADC), matching condition specification (APAAELMMCON[1:0] or APAAELMMCOFF[1:0]) should use “>” or “<“ (because “==” specification might not have matches as expected).

22.5.7.3 Event Input Interval

If the same event input is used for exception processing of multiple elements, the input must have intervals of 1 shared cycle (pclk * 20) or longer. An interval shorter than this length might cause the latter pulse (event input) to be lost.

22.5.7.4 Operation Procedure

In pulse generation using APA, follow the procedure below for generic register setting and element setting. Steps 3 to 8 must be done when the reference bus and event bus are stopped and all pulse generation channels are stopped. (Otherwise unintended operation might be made.)

Table 22.65 Standard Setting and Operation Procedure

Step	Setting Item	Example of Setting and Operation	Remarks
1	a. Reference bus setting	Setting of registers APAA0RFMX0 to sw	
	b. Event bus setting	Setting of registers APAA0EVSL00 to 15, APAA0EVSW, and APAA0EVSC	
	c. Pulse generation channel setting	Setting of registers APAA0CCGAn, APAA0CCGBn	“n” on the left stands for a channel number to be used (0 to 15). All the pulse generation channels to be used are set.
	d. Element RAM setting	Setting of registers APAA0ELMAn, APAA0ELMBn, APAA0ELMCn	“n” on the left stands for the element number to be used (000 to 127). All the elements to be used are set.
2	a. Enabling operation of the reference bus and the event bus	Register APAA0EN = 1	
3	a. Enabling operation of pulse generation channels	Setting of register APAA0CHEN	Setting value = **** (write_mask)_ ****H (channel to be enabled). Set value 1 in the bit that corresponds to the channel to be enabled, and set value 1 in the corresponding bit of write_mask.
4	a. Starting matching comparison	—	Waiting for a Start exception **Note:** When APAA0CCGBn.APAACHNOPSTR is 0, operation starts after a Start exception occurs. A maximum of 20 pclk cycles of wait time are inserted between the event input and operation start.
5	a. Execution of matching comparison	—	Execution of an element
6	a. Ending matching comparison	—	Completion of element execution or suspension due to a Stop exception
7	a. Stopping channel operation	Setting of register APAA0CHEN	Setting value = **** (write_mask)_ ****H (channel to be enabled). Set value 0 in the bit that corresponds to the channel to be enabled, and set value 1 in the corresponding bit of write_mask.
8	a. Confirming stopping of channel operation	Reading of APAA0CSTAn	Stops all pulse generation channels (7 above), waits for 40 pclk or longer, and then read APAA0CSTAn.APAACMPSTAT[4:0] to confirm that the matching comparator is in the disable state.
	b. Stopping operation of the reference bus and the event bus	Register APAA0EN=0	Confirms that all the pulse generation channels are stopped (8-a above) and then sets register APAA0EN to 0.

22.5.7.5 Handling of Unused Exceptions

If any exception should be set so that it is left unused in channel setting or element setting, take measures 1 and 2 described below.

(1) Measure 1: Assigning Unused Exception to Event Input

Handle the unused exception as described below in order to avoid unintended edge detection and level detection.

- Use one of the 64 event input channels (58 external event input channels and six software event input channels) for handling unused exceptions.
- Use one of the 16 event input selector channels exclusively for handling unused exceptions.

NOTE

For the exception input by level (postpone), take care of the level to be enabled when handling unused exception.

(2) Measure 2: Assigning Unused Exception Processing to Event Input for Handling Unused Exceptions

Set the inverted logic of the event input (one channel) and event input selector (one channel), which are assigned using measure 1 above, for the unused exception.

Table 22.66 Setting Example for Avoiding Unintended Exception Processing

Step	Setting Item	Register	Setting Value	Remarks
1	a. Event bus setting	APAA0EVSL12	0000 0039_H	Assign external event source 57 to selected event input 12.
	b. Element RAM setting	APAA0ELMA10	301C $0A00_H$	Reference 3, absolute designation, condition = "< $0A00_H$ " Last element, element continuation
		APAA0ELMB10	001A $0B00_H$	Reference 3, absolute designation, condition = "> $0B00_H$ "
		APAA0ELMC10	7F1C $1C3C_H$	The following is executed as exception processing: • Transfer: External event source 57, rise • Force: External event source 57, rise • Postpone: External event source 57, high → The above 3 exceptions can be disabled.

Note: The table above shows setting examples in cases where Transfer, Force, and Postpone exceptions are not used for element 10. They are based on the standard procedure described in **Section 22.5.7.4, Operation Procedure**.

22.5.7.6 Notes on Interrupt Outputs

On the condition shown below, an interrupt output is continuously generated twice, which results in a single interrupt output. Extra care should be taken in using the setting shown below.

Item	Description
Setting	APAACHNINTSEL[2:0] in APAA0CCGA register are set to 100_B (interrupt output upon either ON match or OFF match).
Condition	Force exception (forced OFF match option) is accepted in 1 pclk cycle after ON match generation, and then OFF match is generated.
Operation	Interrupt output is continuously generated, which results in a single interrupt output.
Remark	Cause of the above operation is that the interrupt generator that generates an interrupt upon either ON match or OFF match is configured so that it generates an interrupt output by performing OR between the ON match interrupt and OFF match interrupt.

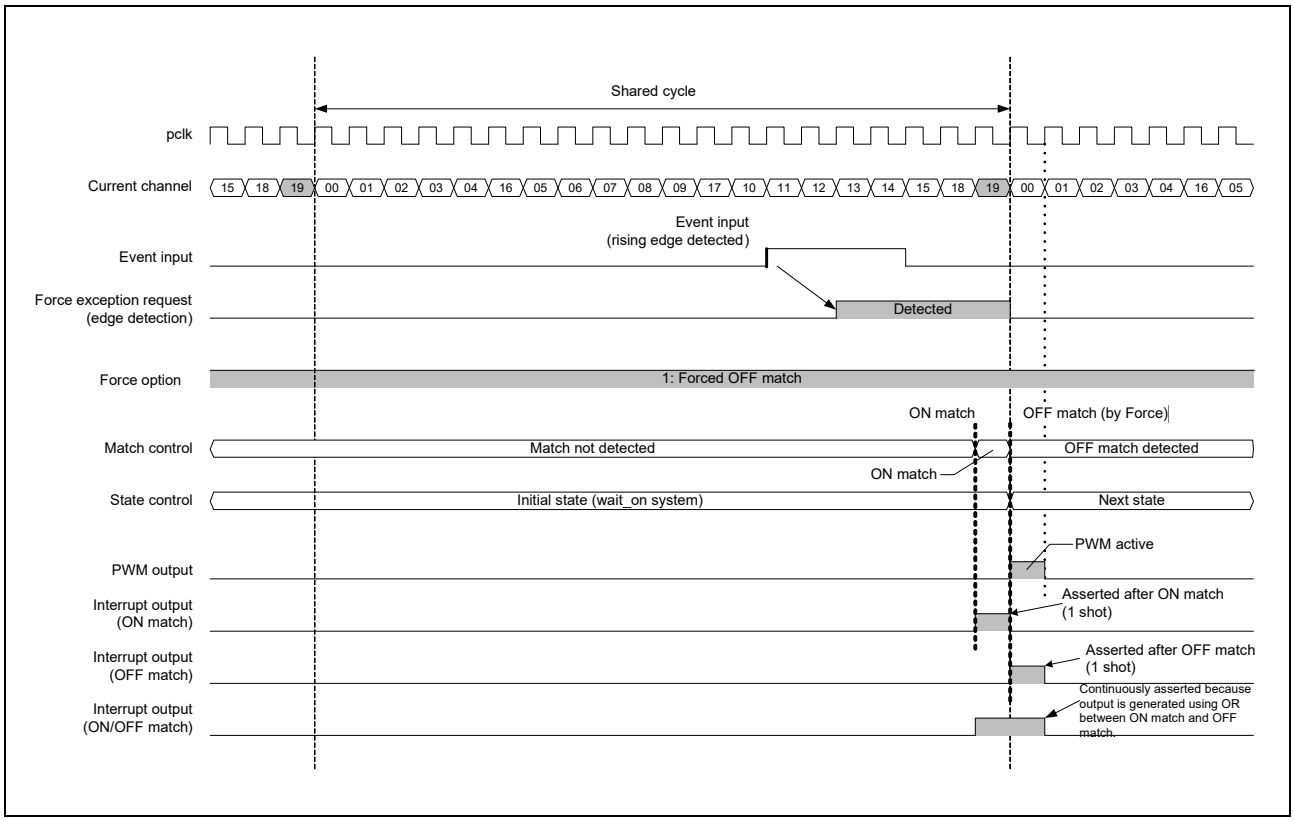


Figure 22.55 Continuous Generation of ON Match Interrupt and OFF Match Interrupt

22.5.8 APA Input Selector

The following sections describe the APA input selector.

22.5.8.1 Block Diagram

Figure 22.56 shows a block diagram of the APA input selector.

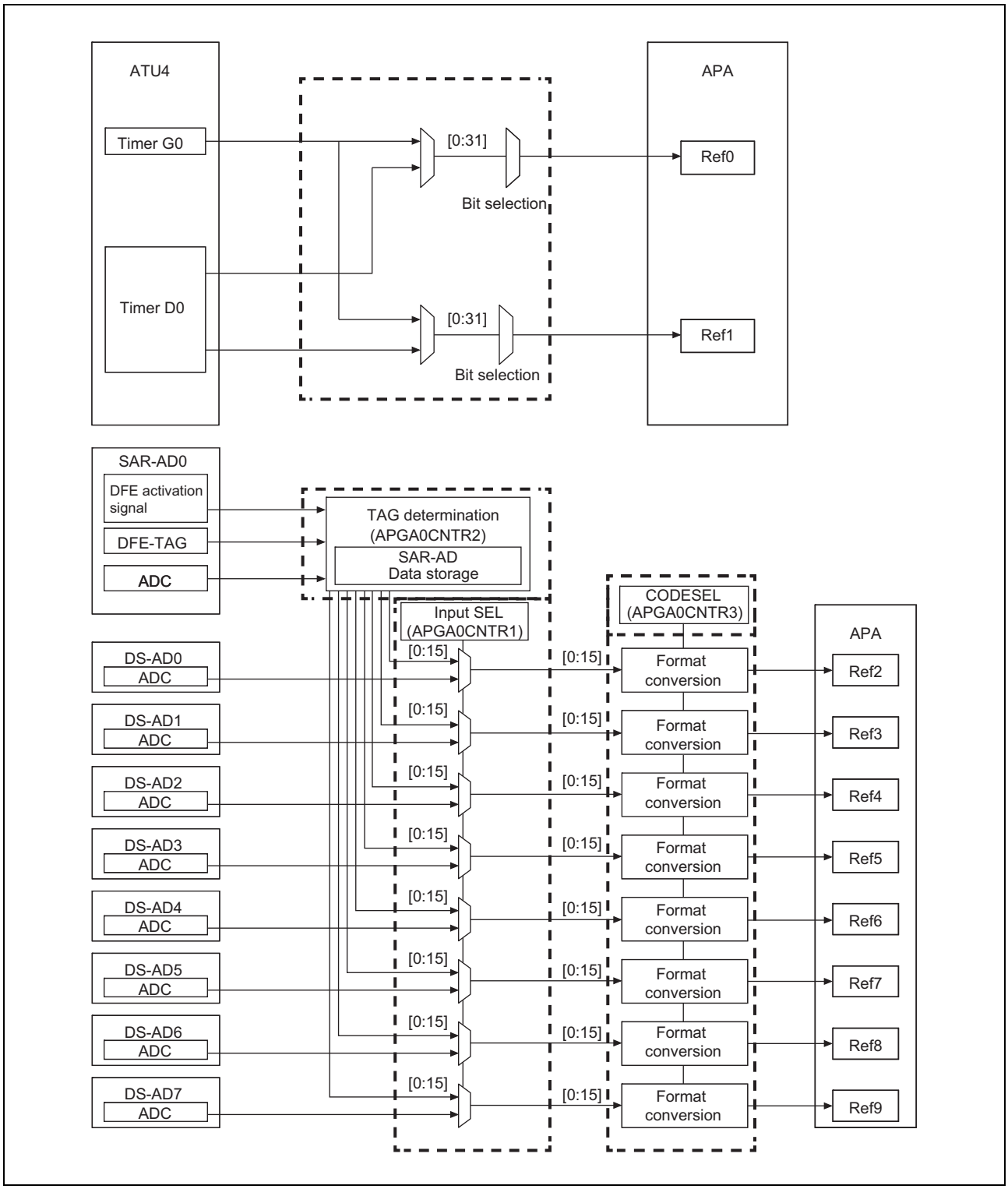


Figure 22.56 Block Diagram of APA Input Selector

22.5.8.2 List of Registers

The addresses of the APA input selector registers are given as offsets from the base address FFED $C000_H$. **Table 22.67** lists the registers.

Table 22.67 APA Input Selector Registers

Offset Address	Symbol	Register Name	Value after Reset	Access Size	Function
0000_H	APGA0CNTR0	APGA0 register control register 0	00_H	8	Selects the counters and bit shift types for ATU-IV to APA connection.
0004_H	APGA0CNTR1	APGA0 register control register 1	00_H	8	Selects DS-AD or SAR-AD0 for reference channels 2 to 5 from AD.
0005_H	APGA0CNTR3	APGA0 register control register 3	FF_H	8	Selects whether to convert the format.
0008_H	APGA0CNTR2	APGA0 register control register 2	$0000\ 0000_H$	32	Selects the channels used for SAR-AD0 (DFE-TAG).

22.5.8.3 Function Details

The following sections describe the functions of the APA input selector modules in detail.

(1) APA Reference Data Selection Control Module

The following describes the functions of the APA reference data selection control module.

- Selects the counter and bit shift types for ATU-IV to APA connection.
- Selects the counter to be output to APA and 16 bits out of the selected counter (32 bits).
- Selects the ATU4 counter to be output to APA reference channels 0 and 1.

Selects either of the following counters for reference channel 0.

- TCNTG0 of ATU-IV timer G
- TCNT1D0 of ATU-IV timer D

Selects either of the following counters for reference channel 1.

- TCNTG0 of ATU-IV timer G
- TCNT2D0 of ATU-IV timer D

The value of each selected counter can be shifted within a range of 0 to 5 bits. 8- and 16-bit shifting is also available.

With APGA0 register control register 0 (APGA0CNTR0), the ATU-IV counter type to be used can be selected. Modifying the register is prohibited while APA is operating.

(a) APGA0CNTR0 — APGA0 Register Control Register 0

Bit	7	6	5	4	3	2	1	0
	REF_SEL1	BIT_SEL1			REF_SEL0	BIT_SEL0		
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 22.68 APGA0CNTR0 Register Contents

Bit Position	Bit Name	Function
7	REF_SEL1	Selects the counter to be output to the APA reference channel 1. 0: TCNTG0 of ATU timer G 1: TCNT2D0 of ATU timer D
6 to 4	BIT_SEL1	Selects 16 bits to be output to APA reference channel 1. 000: No shift 001: 1-bit shift 010: 2-bit shift 011: 3-bit shift 100: 4-bit shift 101: 5-bit shift 110: 8-bit shift (middle 16 bits) 111: 16-bit shift (upper 16 bits)
3	REF_SEL0	Selects the counter to be output to the APA reference channel 0. 0: TCNTG0 of ATU timer G 1: TCNT1D0 of ATU timer D
2 to 0	BIT_SEL0	Selects 16 bits to be output to APA reference channel 0. 000: No shift 001: 1-bit shift 010: 2-bit shift 011: 3-bit shift 100: 4-bit shift 101: 5-bit shift 110: 8-bit shift (middle 16 bits) 111: 16-bit shift (upper 16 bits)

(2) AD Reference Selection

For APA reference channels 2 to 9 from AD, either DS-AD or SAR-AD0 can be selected and used. The format of the input data is converted. When SAR-AD0 is used, the DFE TAG used by the DFE is read to determine the channels to be used and the data of the specified channels is retained and output.

- Modifying the reference selection is prohibited while APA is operating

(a) APGA0CNTR1 — APGA0 Register Control Register 1

APGA0 register control register 1 (APGA0CNTR1) selects the type of AD to be used.

Bit	7	6	5	4	3	2	1	0
	REF_SEL9	REF_SEL8	REF_SEL7	REF_SEL6	REF_SEL5	REF_SEL4	REF_SEL3	REF_SEL2
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

CAUTION

Since SAR-AD0 cannot be used as APA reference in this product, use this register with its value after a reset, 00_H.

Table 22.69 APGA0CNTR1 Register Contents

Bit Position	Bit Name	Function
7	REF_SEL9	Selects AD data to be input to APA reference channel 9. 0: DS-AD7 1: Data with TAG identical to SAR-AD0 TAG_SEL9
6	REF_SEL8	Selects AD data to be input to APA reference channel 8. 0: DS-AD6 1: Data with TAG identical to SAR-AD0 TAG_SEL8
5	REF_SEL7	Selects AD data to be input to APA reference channel 7. 0: DS-AD5 1: Data with TAG identical to SAR-AD0 TAG_SEL7
4	REF_SEL6	Selects AD data to be input to APA reference channel 6. 0: DS-AD4 1: Data with TAG identical to SAR-AD0 TAG_SEL6
3	REF_SEL5	Selects AD data to be input to APA reference channel 5. 0: DS-AD3 1: Data with TAG identical to SAR-AD0 TAG_SEL5
2	REF_SEL4	Selects AD data to be input to APA reference channel 4. 0: DS-AD2 1: Data with TAG identical to SAR-AD0 TAG_SEL4
1	REF_SEL3	Selects AD data to be input to APA reference channel 3. 0: DS-AD1 1: Data with TAG identical to SAR-AD0 TAG_SEL3
0	REF_SEL2	Selects AD data to be input to APA reference channel 2. 0: DS-AD0 1: Data with TAG identical to SAR-AD0 TAG_SEL2

(b) APGA0CNTR2 — APGA0 Register Control Register 2

APGA0 register control register 2 (APGA0CNTR2) specifies the DEF-TAG and indirectly selects the channels to be used for SAR-AD0. The channels to be used for SAR-AD0 are determined based on the DFE-TAG and DFE activation signal. Note that this register does not directly select the channels.

Modifying the register is prohibited while APA is operating.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	SARAD0_TAG_SEL9				SARAD0_TAG_SEL8				SARAD0_TAG_SEL7				SARAD0_TAG_SEL6			
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	SARAD0_TAG_SEL5				SARAD0_TAG_SEL4				SARAD0_TAG_SEL3				SARAD0_TAG_SEL2			
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 22.70 APGA0CNTR2 Register Contents

Bit Position	Bit Name	Function
31 to 28	SARAD0_TAG_SEL9	Selects SAR-AD0 data to be used when REF_SEL9 = 1. Data with DFTAG identical to the specified value is selected.
27 to 24	SARAD0_TAG_SEL8	Selects SAR-AD0 data to be used when REF_SEL8 = 1. Data with DFTAG identical to the specified value is selected.
23 to 20	SARAD0_TAG_SEL7	Selects SAR-AD0 data to be used when REF_SEL7 = 1. Data with DFTAG identical to the specified value is selected.
19 to 16	SARAD0_TAG_SEL6	Selects SAR-AD0 data to be used when REF_SEL6 = 1. Data with DFTAG identical to the specified value is selected.
15 to 12	SARAD0_TAG_SEL5	Selects SAR-AD0 data to be used when REF_SEL5 = 1. Data with DFTAG identical to the specified value is selected.
11 to 8	SARAD0_TAG_SEL4	Selects SAR-AD0 data to be used when REF_SEL4 = 1. Data with DFTAG identical to the specified value is selected.
7 to 4	SARAD0_TAG_SEL3	Selects SAR-AD0 data to be used when REF_SEL3 = 1. Data with DFTAG identical to the specified value is selected.
3 to 0	SARAD0_TAG_SEL2	Selects SAR-AD0 data to be used when REF_SEL2 = 1. Data with DFTAG identical to the specified value is selected.

(3) Conversion of AD Reference Format

Although it is assumed that signed reference data is input from each AD, the APA only handles unsigned data for comparison. Therefore, it is necessary to convert the format of the reference data input from each AD as shown below.

- Hardware inverts the MSB value.

AD Output		APA Input	
Hexadecimal	**Decimal**	**Hexadecimal**	**Decimal**
$7FFF_H$	+32767	$FFFF_H$	65535
|	|	|	|
0001_H	1	8001_H	32769
0000_H	0	8000_H	32768
$FFFF_H$	-1	$7FFF_H$	32767
|	|	|	|
8000_H	-32768	0000_H	0

CAUTION

Zero is always input as a sign of data from SAR-AD (it is necessary to inform the user that the data format should be converted as required). Therefore, in setting elements in the APA, the MSB must always be set in such a manner that the value should be 8000_H or greater.

(a) APGA0CNTR3 — APGA0 Register Control Register 3

Bit	7	6	5	4	3	2	1	0
	CODE_SEL9	CODE_SEL8	CODE_SEL7	CODE_SEL6	CODE_SEL5	CODE_SEL4	CODE_SEL3	CODE_SEL2
Value after reset	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 22.71 APGA0CNTR3 Register Contents

Bit Position	Bit Name	Function
7	CODE_SEL9	Selects whether to convert a sign of AD data to be input to APA Ref9. 0: Sign is not converted. 1: Sign is converted.
6	CODE_SEL8	Selects whether to convert a sign of AD data to be input to APA Ref8. 0: Sign is not converted. 1: Sign is converted.
5	CODE_SEL7	Selects whether to convert a sign of AD data to be input to APA Ref7. 0: Sign is not converted. 1: Sign is converted.
4	CODE_SEL6	Selects whether to convert a sign of AD data to be input to APA Ref6. 0: Sign is not converted. 1: Sign is converted.
3	CODE_SEL5	Selects whether to convert a sign of AD data to be input to APA Ref5. 0: Sign is not converted. 1: Sign is converted.
2	CODE_SEL4	Selects whether to convert a sign of AD data to be input to APA Ref4. 0: Sign is not converted. 1: Sign is converted.
1	CODE_SEL3	Selects whether to convert a sign of AD data to be input to APA Ref3. 0: Sign is not converted. 1: Sign is converted.
0	CODE_SEL2	Selects whether to convert a sign of AD data to be input to APA Ref2. 0: Sign is not converted. 1: Sign is converted.

22.5.8.4 Overview of Operations

(1) APA Reference Data Selection Control Module

Figure 22.57 shows a block diagram of the APA reference data selection control module.

Channels 1 and 2 of timer G and timer D input from the ATU-IV timer are controlled using the APGA0CNTR0 register to select the ATU-IV counter to be output to APA reference channel 0 and reference channel 1.

Select the bits considering that the resolution of the reference input to be higher than the count cycle of the counter to be connected.

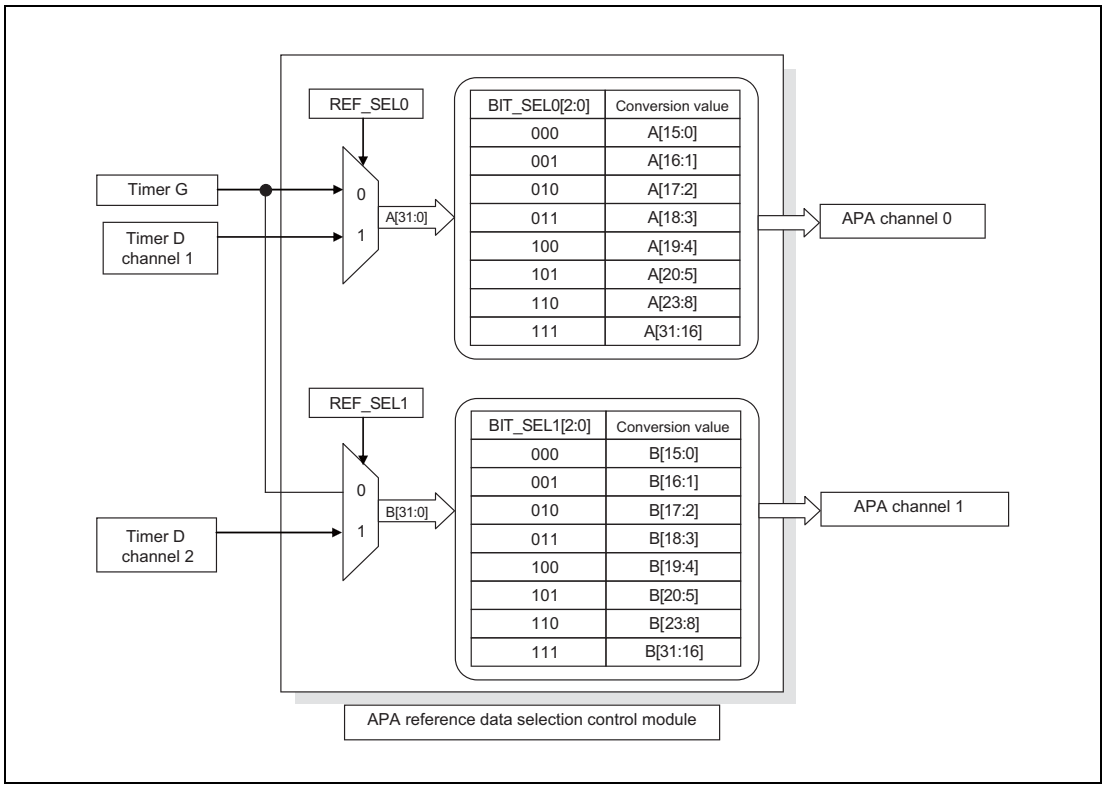


Figure 22.57 Block Diagram of APA Reference Data Selection Control Module

(2) AD Reference Selection

Figure 22.58 and **Figure 22.59** show block diagrams of the APA reference selection functions.

A match between the values set in SARAD0_TAG_SEL9 to SARAD0_TAG_SEL2 in the APGA0CNTR2 register and the AD tag signal (ADC0DFTAG) input from PB0 are checked and confirmed, and which signals to be output as addf_dat9 to addf_dat2 is selected.

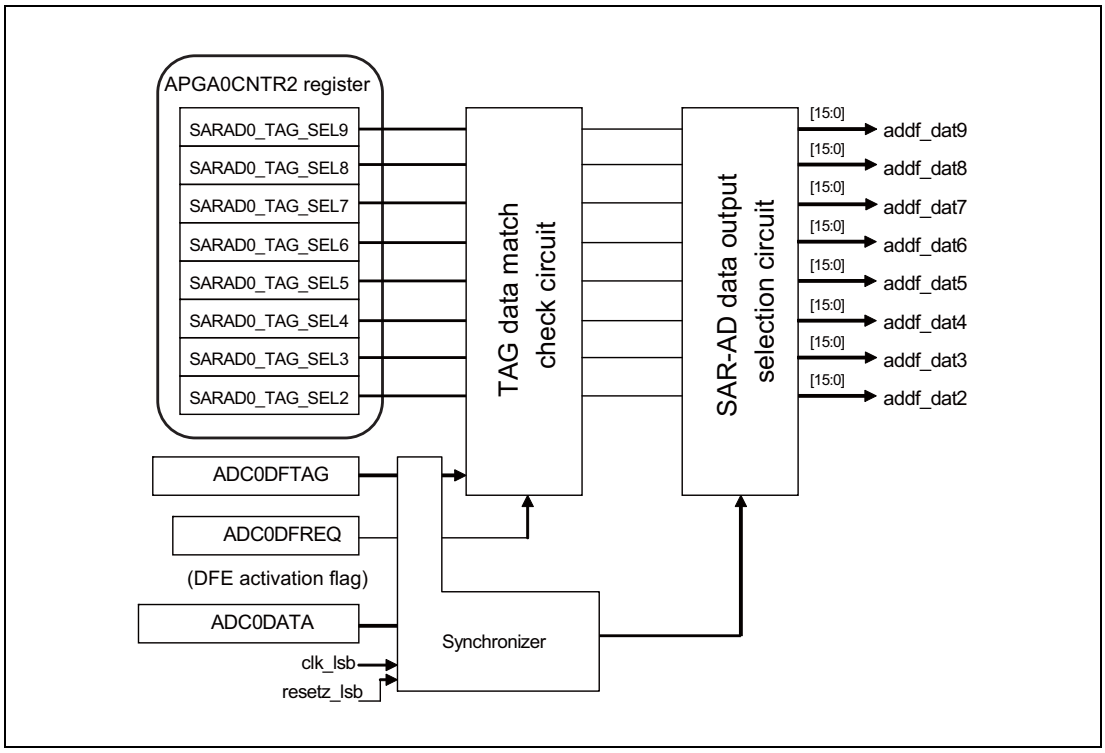


Figure 22.58 Block Diagram of SAR-AD Data Selector

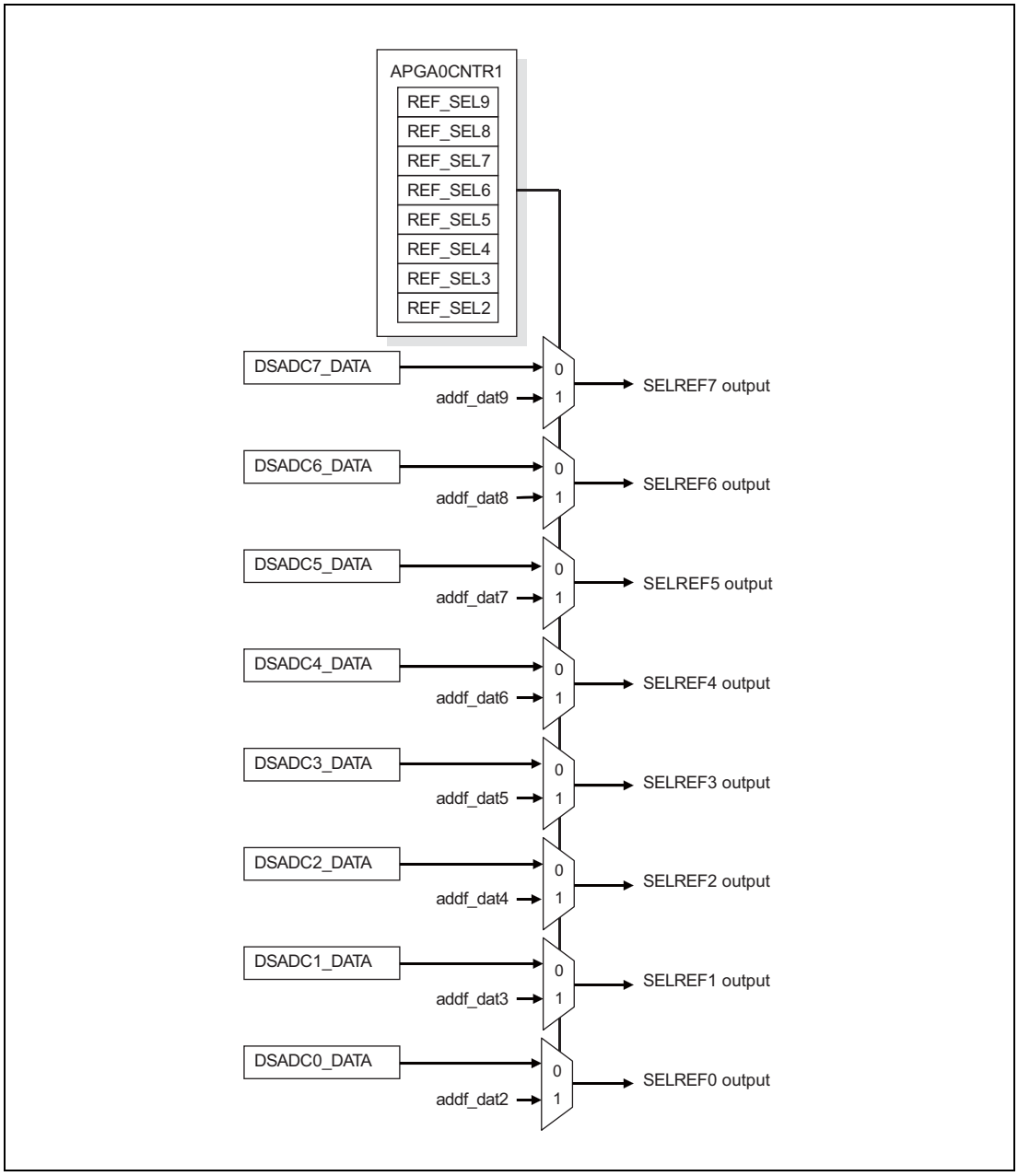


Figure 22.59 Block Diagram of SAR-AD and DA-AD Selector

(3) Conversion of AD Reference Format

Figure 22.60 shows a block diagram of the APA reference format conversion function.

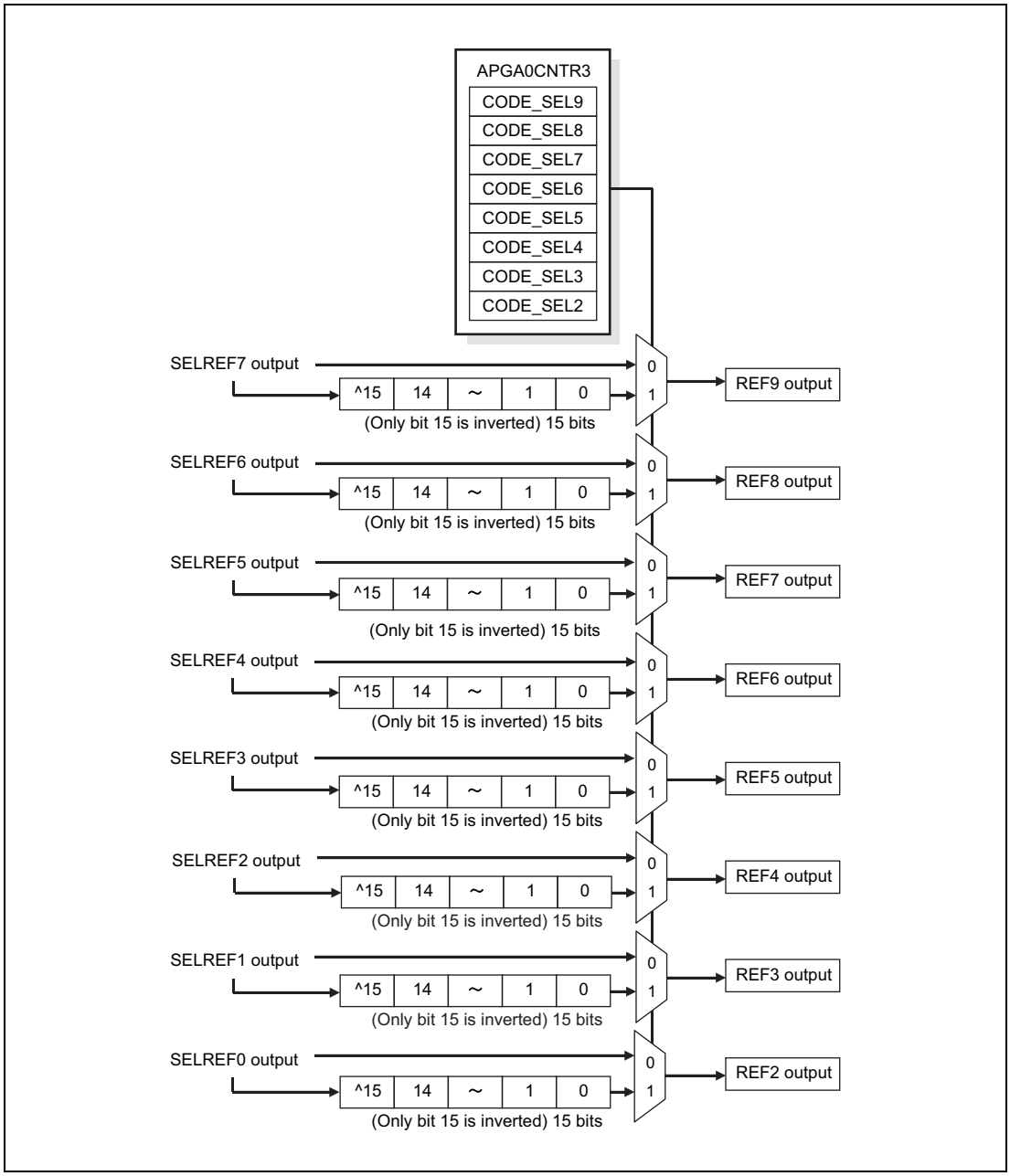


Figure 22.60 Block Diagram of APA Reference Format Conversion Function

Section 23 Motor Control Timer (TSG2)

23.1 Functions of TSG2n

- Channels
  This microcontroller provides two channels of TSG2n (n = 0, 1).

Table 23.1 TSG2 Channels

	TSG2
Number of Channels	2
Name	TSG20, TSG21

Meaning of n

Throughout this section, the TSG2 channels are identified by the index n (n = 0 or 1): for example, TSG2n control register 0 (TSG2nCTL0).

- Register addresses
  TSG2n register addresses are given as offsets from the individual base addresses <TSG2n_base>. **Table 23.2** shows the base addresses of TSG2n.

Table 23.2 TSG2n Register Base Address

TSG2n	<TSG2n_base> Address
TSG20	FFE7 0000_H
TSG21	FFE7 1000_H

- Interrupt requests
  The TSG2n interrupt requests are listed in **Table 23.3**.

Table 23.3 List of TSG2n Interrupt Requests

TSG2n Interrupt Request	Function	Connected to
INTTSG2nI[12:0]	TSG2n compare match interrupt 0 to 12	Interrupt controller (INTC)
INTTSG2nIPEK	TSG2n peak interrupt	Interrupt controller (INTC)
INTTSG2nIVLY	TSG2n trough interrupt	Interrupt controller (INTC)
INTTSG2nIER	TSG2n error interrupt	Interrupt controller (INTC) PIC1
INTTSG2nIWN	TSG2n warning interrupt	Interrupt controller (INTC)

- DMA transfer requests
  The Table below shows TSG2n DMA transfer requests.

Table 23.4 List of TSG2n DMA Transfer Requests

TSG2n Interrupt Request	Function	Connected to
INTTSG2nI[12:11]	TSG2n compare match interrupt 11 to 12	DMA controller (DTS)
INTTSG2nIPEK	TSG2n peak interrupt	DMA controller (DTS)
INTTSG2nIVLY	TSG2n trough interrupt	DMA controller (DTS)

23.2 Functional Overview

The TSG2n is a 16-bit timer counter with various motor control functions.

- Count clock resolution: Minimum 12.5 ns (count clock = 80 MHz)
- Operating mode corresponding to various motor control methods
- Compare registers with reload buffer
- 10-bit dead time counter
  - Dead time counter with reload buffer
  - Independent dead time can be set for positive to inverse phase change and inverse to positive phase change.
- A/D conversion trigger signal generation
  - Three A/D conversion trigger signals can be generated by the compare registers TSG2nDCMP0W and TSG2nDCMP2.
  - Skipping function of A/D conversion trigger signals TSTADT0 and TSTADT1 can be set independently. The skipping ratio can be selected among 1/1, 1/2, 1/4, and 1/8.
  - The dedicated pin (TSOn7) can be used to output the toggle or diagnostic signal set by the TSTADT0 signal and reset by the TSTADT1 signal.
- Interrupt skipping
  - Skipping rate: 1/1 to 1/32
- Forced output stop function
  - Using the TOP function allows the high impedance control of the TSOn1 to TSOn6 pin output.
- Compare value setting
  - Reload (simultaneous rewrite) or anytime rewrite can be selected.
- Reload mode
  - Writing to TSG2nCMP1 enables reload (the reload request flag (TSG2nRSF) is set), and allows simultaneous transfer of the values of multiple registers.
  - Data can be transferred at peak/trough/peak or trough reload timing
  - Reload request flag (TSG2nRSF)
  - Register address assignment allowing DMA transfer
  - Reload skipping
- HT-PWM mode
  - 0 to 100% PWM duty cycle output is possible (with possible dead time reduction).
  - The LSB in the compare register can be used to append an additional pulse to the PWM output during count up, thus improving the output resolution without extra load on software.
- 120-DC control
  - A trigger signal can be generated by an offset in conjunction with the semi-automatic drive function.
- Three-phase encoder function (hall sensor signals can be input).

- Active level of the output pins (TSOn1 to TSOn6) can be set individually.
- Fail-safe function (warning interrupt or error interrupt can be generated)
  - Simultaneous active output detection function for positive and inverse phases.
  - Abnormal input detection function of the three-phase encoder

23.3 Configuration

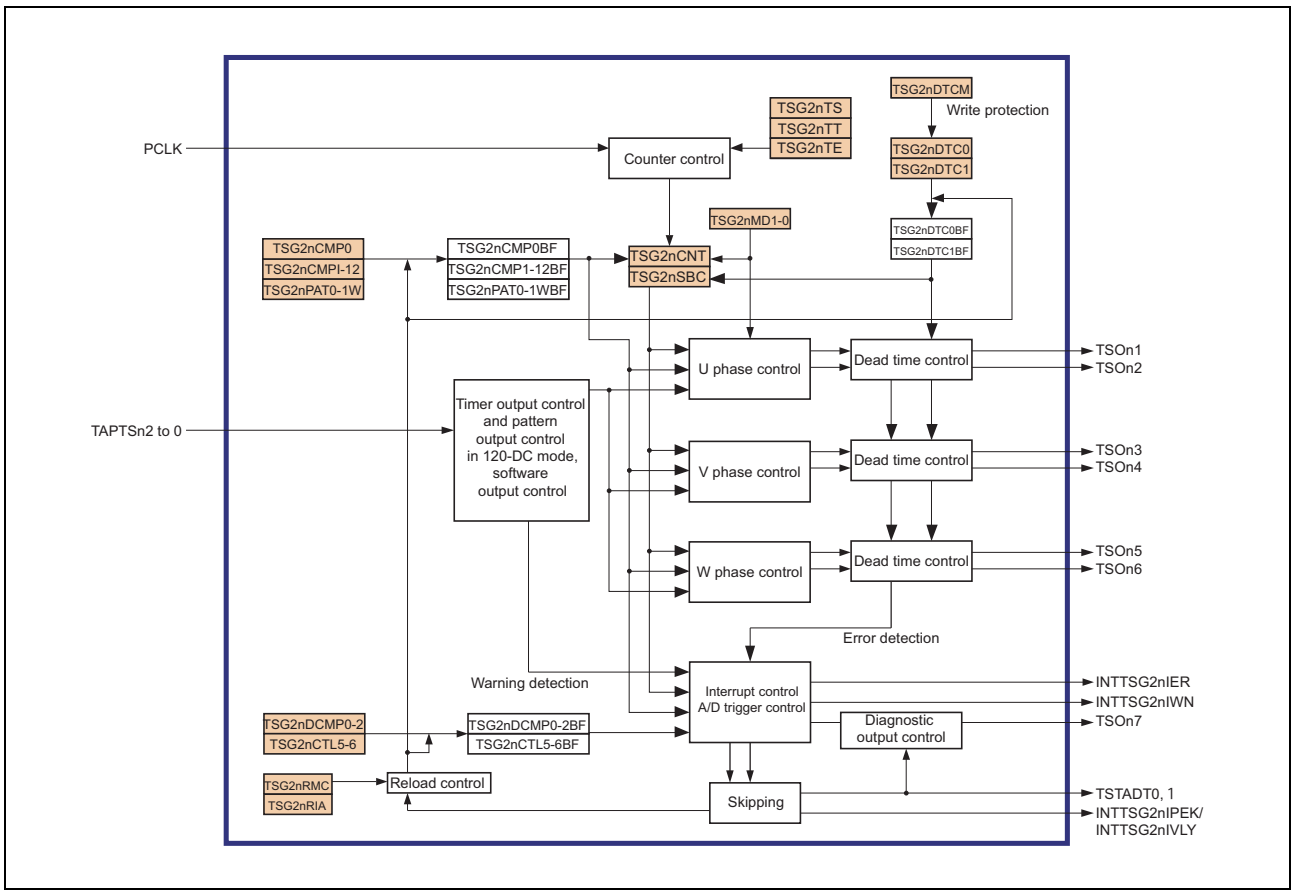


Figure 23.1 Block Diagram of TSG2n

23.4 Registers

This section contains a description of all registers of the TSG2n (n = 0, 1).

23.4.1 List of Registers

The TSG2n (n = 0, 1) registers and addresses are shown in **Table 23.5**.

The base address of TSG20 is FFE7 0000_H and TSG21 is FFE7 1000_H respectively.

For the actual addresses, the offset values shown in the table below are added to the base addresses.

Table 23.5 TSG2n (n = 0, 1) Registers (1/2)

Register Name	Function	R/W	Reset value	Access			Offset address	Dynamic	Reload
				8	16	32			
TSG2nIOC2	TSG2n I/O control register 2	R/W	0000_H	×	√	×	000_H	Enable	Disable
TSG2nCTL3	TSG2n control register 3	R/W	00_H	√	×	×	004_H	Enable	Disable
TSG2nCTL5	TSG2n control register 5	R/W	0000_H	×	√	×	008_H	Enable	Disable
TSG2nCTL6	TSG2n control register 6	R/W	0000_H	×	√	×	$00C_H$	Enable	Disable
TSG2nSTR0	TSG2n status register 0	R	00_H	√	×	×	010_H	Enable	Disable
TSG2nSTR1	TSG2n status register 1	R	00_H	√	×	×	014_H	Enable	Disable
TSG2nSTR2	TSG2n status register 2	R	0000_H	×	√	×	018_H	Enable	Disable
TSG2nSTC	TSG2n status clear trigger register	W	0000_H	×	√	×	$01C_H$	Enable	Disable
TSG2nOPT0	TSG2n option register 0	R/W	00_H	√	×	×	020_H	Enable	Disable
TSG2nOPT1	TSG2n option register 1	R/W	00_H	√	×	×	024_H	Enable	Disable
TSG2nCNT	TSG2n timer counter register	R	0000_H	×	√	×	028_H	Enable	Disable
TSG2nSBC	TSG2n timer sub-counter register	R	0000_H	×	√	×	$02C_H$	Enable	Disable
TSG2nTRG0	TSG2n trigger register 0	W	00_H	√	×	×	030_H	Enable	Disable
TSG2nTRG1	TSG2n trigger register 1	W	00_H	√	×	×	034_H	Enable	Disable
TSG2nCMP1W	TSG2n compare registers 1,2	R/W	$0000\ 0000_H$	×	×	√	040_H	Enable	Enable
TSG2nCMP5W	TSG2n compare registers 5,6	R/W	$0000\ 0000_H$	×	×	√	044_H	Enable	Enable
TSG2nCMP9W	TSG2n compare registers 9,10	R/W	$0000\ 0000_H$	×	×	√	048_H	Enable	Enable
TSG2nCMP3W	TSG2n compare registers 3,4	R/W	$0000\ 0000_H$	×	×	√	$04C_H$	Enable	Enable
TSG2nCMP7W	TSG2n compare registers 7,8	R/W	$0000\ 0000_H$	×	×	√	050_H	Enable	Enable
TSG2nCMP11W	TSG2n compare registers 11,12	R/W	$0000\ 0000_H$	×	×	√	054_H	Enable	Enable
TSG2nCMP0	TSG2n compare register 0	R/W	$0000\ 0000_H$	×	×	√	058_H	Enable	Enable
TSG2nDCMP0W	TSG2n diagnostic compare registers 0,1	R/W	$0000\ 0000_H$	×	×	√	$05C_H$	Enable	Enable
TSG2nDCMP2	TSG2n diagnostic compare register 2	R/W	$0000\ 0000_H$	×	×	√	060_H	Enable	Enable
TSG2nPAT0W	TSG2n pattern register 0	R/W	$0000\ 0000_H$	×	×	√	064_H	Enable	Enable
TSG2nPAT1W	TSG2n pattern register 1	R/W	$0000\ 0000_H$	×	×	√	068_H	Enable	Enable
TSG2nDTC0W	TSG2n dead time compare register 0	R/W	$0000\ 0000_H$	×	×	√	$06C_H$	Enable	Enable
TSG2nDTC1W	TSG2n dead time compare register 1	R/W	$0000\ 0000_H$	×	×	√	070_H	Enable	Enable
TSG2nIOC3	TSG2n I/O control register 3	R/W	$0000\ 0000_H$	×	×	√	074_H	Enable	Enable
TSG2nCTL4	TSG2n control register 4	R/W	$0000\ 0000_H$	×	×	√	$07C_H$	Enable	Enable
TSG2nCMP1	TSG2n compare register 1	R/W	0000_H	×	√	×	080_H	Enable	Enable
TSG2nCMP2	TSG2n compare register 2	R/W	0000_H	×	√	×	084_H	Enable	Enable
TSG2nCMP5	TSG2n compare register 5	R/W	0000_H	×	√	×	088_H	Enable	Enable
TSG2nCMP6	TSG2n compare register 6	R/W	0000_H	×	√	×	$08C_H$	Enable	Enable
TSG2nCMP9	TSG2n compare register 9	R/W	0000_H	×	√	×	090_H	Enable	Enable
TSG2nCMP10	TSG2n compare register 10	R/W	0000_H	×	√	×	094_H	Enable	Enable
TSG2nCMP3	TSG2n compare register 3	R/W	0000_H	×	√	×	098_H	Enable	Enable

Table 23.5 TSG2n (n = 0, 1) Registers (2/2)

Register Name	Function	R/W	Reset value	Access			Offset address	Dynamic	Reload
				8	16	32			
TSG2nCMP4	TSG2n compare register 4	R/W	0000_H	×	√	×	$09C_H$	Enable	Enable
TSG2nCMP7	TSG2n compare register 7	R/W	0000_H	×	√	×	$0A0_H$	Enable	Enable
TSG2nCMP8	TSG2n compare register 8	R/W	0000_H	×	√	×	$0A4_H$	Enable	Enable
TSG2nCMP11	TSG2n compare register 11	R/W	0000_H	×	√	×	$0A8_H$	Enable	Enable
TSG2nCMP12	TSG2n compare register 12	R/W	0000_H	×	√	×	$0AC_H$	Enable	Enable
TSG2nCMPU	TSG2n U phase compare register	R/W	0000_H	×	√	×	$0B0_H$	Enable	Enable
TSG2nCMPV	TSG2n V phase compare register	R/W	0000_H	×	√	×	$0B4_H$	Enable	Enable
TSG2nCMPW	TSG2n W phase compare register	R/W	0000_H	×	√	×	$0B8_H$	Enable	Enable
TSG2nUPW	TSG2n U phase period register	R/W	0000_H	×	√	×	$0BC_H$	Enable	Enable
TSG2nVPW	TSG2n V phase period register	R/W	0000_H	×	√	×	$0C0_H$	Enable	Enable
TSG2nWPW	TSG2n W phase period register	R/W	0000_H	×	√	×	$0C4_H$	Enable	Enable
TSG2nIOC0	TSG2n I/O control register 0	R/W	$7E_H$	√	×	×	200_H	Disable	Disable
TSG2nIOC1	TSG2n I/O control register 1	R/W	00_H	√	×	×	204_H	Disable	Disable
TSG2nCTL0	TSG2n control register 0	R/W	00_H	√	×	×	208_H	Disable	Disable
TSG2nCTL1	TSG2n control register 1	R/W	0000_H	×	√	×	$20C_H$	Disable	Disable
TSG2nDTPR	TSG2n dead time protection register	R/W	0000_H	×	√	×	210_H	Disable	Disable

- Dynamic: “Enable” means that registers can be read during TSG2 operation. “Disable” means that the contents of registers cannot be changed during TSG2 operation.
- Reload: “Reload” means that the register is provided with the reload function.

23.4.2 TSG2n Register Details

23.4.2.1 TSG2nCTL0 — TSG2n Control Register 0

This register specifies the pulse width for TSOn7 and operating mode of the TSG2n.

This register can be written only when TSG2nTE = 0 (while macros are stopped).

Access: This register can be read/written in 8-bit units.

Address: TSG20: FFE7 0208_H
TSG21: FFE7 1208_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	TSG2n DWD	—	—	TSG2nMD[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R	R	R/W	R/W

Table 23.6 TSG2nCTL0 Register Contents

Bit Position	Bit Name	Functions
7 to 5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4	TSG2nDWD	TSOn7 Pulse Width Control 0: The TSOn7 pulse width is 8 clock cycles. 1: The TSOn7 pulse width is 16 clock cycles. This bit controls the pulse width of TSOn7 and is valid when TSG2nTGS = 1. (See the description of TSG2nTGS) For the detailed functions, refer to **Section 23.5.4.1 (1), TSOn7 Pin Output Control**.
3, 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1, 0	TSG2nMD[1:0]	TSG2n Mode Control 00: PWM mode 01: High-accuracy triangle-wave PWM mode 10: Shift pulse PWM mode 11: 120-degree energization control mode These bits are mode control bits of TSG2n. TSG2n has four modes. PWM mode controls six PWM timer outputs. Each timer output is controlled by the two compare registers (set and clear). High-accuracy triangle-wave PWM mode (HT-PWN mode) controls three complementary PWM outputs by using up-count and down-count (triangle-wave count). Shift pulse PWM mode (SP-PWM mode) controls three complementary PWM outputs by using up-count (saw-wave count). 120-degree energization control mode (120-DC mode) controls six PWM outputs by using twelve compare registers, two pattern registers, and six output steps.

23.4.2.2 TSG2nCTL1 — TSG2n Control Register 1

This register controls the error/warning detection and can be written only when TSG2nTE = 0 (while macros are stopped).

Refer to the description of TSG2nSTR2 register and **Section 23.10, Error and Warning Interrupts**.

Access: This register can be read/written in 16-bit units.

Address: TSG20: FFE7 020C$_H$
TSG21: FFE7 120C$_H$

Value after reset: 0000$_H$

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TSG2n TBA2	TSG2n TBA1	TSG2n TBA0	TSG2n PPC	TSG2n PEC	—	TSG2n NDC	TSG2n PRC	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R/W	R/W	R	R

Table 23.7 TSG2nCTL1 Register Contents

Bit Position	Bit Name	Functions
15 to 10	Reserved	When read, the value after reset is read. When writing, write the value after reset.
9	TSG2nTBA2	TSG2nTBF2 Control 0: Disables TSG2nTBF2 1: Enables TSG2nTBF2
8	TSG2nTBA1	TSG2nTBF1 Control 0: Disables TSG2nTBF1 1: Enables TSG2nTBF1
7	TSG2nTBA0	TSG2nTBF0 Control 0: Disables TSG2nTBF0 1: Enables TSG2nTBF0
6	TSG2nPPC	TSG2nPPF Control 0: Disables TSG2nPPF 1: Enables TSG2nPPF
5	TSG2nPEC	TSG2nPEF Control 0: Disables TSG2nPEF 1: Enables TSG2nPEF
4	Reserved	When read, the value after reset is read. When writing, write the value after reset.
3	TSG2nNDC	TSG2nNDF Control 0: Disables TSG2nNDF 1: Enables TSG2nNDF
2	TSG2nPRC	TSG2nPRF Control 0: Disables TSG2nPRF 1: Enables TSG2nPRF
1, 0	Reserved	When read, the value after reset is read. When writing, write the value after reset.

23.4.2.3 TSG2nCTL3 — TSG2n Control Register 3

This register controls TSG2n reload /IMWRITE function.

Access: This register can be read/written in 8-bit units.

Address: TSG20: FFE7 0004_H
TSG21: FFE7 1004_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TSG2n RIA	TSG2n RMC
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 23.8 TSG2nCTL3 Register Contents

Bit Position	Bit Name	Function
7 to 2	Reserved	When read, the value after reset is read. When writing, write the value after reset.
1	TSG2nRIA	Reload and INTTSG2nIPEK/INTTSG2nIVLY Adjustment 0: The reload timing is controlled only by TSG2nPRE or TSG2nVRE. 1: The reload timing is adjusted by INTTSG2nIPEK or INTTSG2nIVLY output timing. TSG2nRIA controls reload timing and is valid when TSG2nRMC = 0.
0	TSG2nRMC	Register Mode Control 0: Update the value of registers with reload function when reloading 1: Update the value of registers when written. This function is called “Reload” when TSG2nRMC = 0 and “IMWRITE” (immediate write) when TSG2nRMC = 1

23.4.2.4 TSG2nCTL4 — TSG2n Control Register 4

This register controls INTTSG2nIPEK/INTTSG2nIVLY interrupt function.

Access: This register can be read/written in 32-bit units.

Address: TSG20: FFE7 007C$_H$
TSG21: FFE7 107C$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TSG2n PRE	TSG2n VRE	TSG2n PIE	TSG2n VIE	TSG2nRCC[04:00]				
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 23.9 TSG2nCTL4 Register Contents

Bit Position	Bit Name	Function
31 to 9	Reserved	When read, the value after reset is read. When writing, write the value after reset.
8	TSG2nPRE	Reloading by INTTSG2nIPEK Timing Enable Bit 0: Disables reload operation at the peak timing (in HT-PWM mode) Disables reload operation at the PWM cycle timing (in any mode other than HT-PWM mode) 1: Enables reload operation at the peak timing (in HT-PWM mode) Enables reload operation at the PWM cycle timing (in any mode other than HT-PWM mode) When setting TSG2nRMC to 0 in any mode other than HT-PWM mode, reloading does not proceed. Accordingly, when recovery from the abnormal state is required, the setting for IMWRITE function needs to be changed.
7	TSG2nVRE	Reloading by INTTSG2nIVLY Timing Enable Bit 0: Disables reload operation at the trough timing 1: Enables reload operation at the trough timing This bit is only valid in HT-PWM mode.
6	TSG2nPIE	INTTSG2nIPEK Interrupt Enable 0: Masks INTTSG2nIPEK output 1: Does not mask INTTSG2nIPEK output
5	TSG2nVIE	INTTSG2nIVLY Interrupt Enable 0: Masks INTTSG2nIVLY output 1: Does not mask INTTSG2nIVLY output This bit is only valid in HT-PWM mode.
4 to 0	TSG2nRCC [04:00]	Reload Skipping Control 00000: No reload/interrupt skipping 00001: 1 skipping/1 generation out of 2 reloads/interrupts 00010: 2 skippings/1 generation out of 3 reloads/interrupts 00011: 3 skippings/1 generation out of 4 reloads/interrupts | 11110: 30 skippings/1 generation out of 31 reloads/interrupts 11111: 31 skippings/1 generation out of 32 reloads/interrupts

23.4.2.5 TSG2nCTL5 — TSG2n Control Register 5

This register controls A/D conversion trigger (TSTADT0) function.

See **Section 23.9, A/D Conversion Trigger Function**.

Access: This register can be read/written in 16-bit units.

Address: TSG20: FFE7 0008_H
TSG21: FFE7 1008_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	TSG2nACC [01:00]		TSG2n AT09	TSG2n AT08	TSG2n AT07	TSG2n AT06	TSG2n AT05	TSG2n AT04	TSG2n AT03	TSG2n AT02	TSG2n AT01	TSG2n AT00
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 23.10 TSG2nCTL5 Register Contents (1/2)

Bit Position	Bit Name	Function
15 to 12	Reserved	When read, the value after reset is read. When writing, write the value after reset.
11, 10	TSG2nACC [01:00]	TSTADT0 Skipping Control 00: No TSTADT0 skipping 01: 1 skipping/1 trigger output out of 2 A/D source triggers 10: 3 skipping/1 trigger output out of 4 A/D source triggers 11: 7 skipping/1 trigger output out of 8 A/D source triggers When a write access is made, the TSTADT0 skipping counter is simultaneously cleared.
9	TSG2nAT09	TSTADT0 Trigger Control 9 0: Not selected 1: Selects TSG2nSBC peak timing as A/D source trigger. This bit is only valid in HT-PWM mode.
8	TSG2nAT08	TSTADT0 Trigger Control 8 0: Not selected 1: Selects TSG2nSBC trough timing as A/D source trigger. This bit is only valid in HT-PWM mode.
7	TSG2nAT07	TSTADT0 Trigger Control 7 0: Not selected 1: Selects the matching points of TSG2nDCMP2 and TSG2nCNT as A/D source triggers for down-counting by TSG2nCNT. This bit is only valid in HT-PWM mode. In other modes, TSG2nCNT does not count down.
6	TSG2nAT06	TSTADT0 Trigger Control 6 0: Not selected 1: Selects the matching points of TSG2nDCMP2 and TSG2nCNT as A/D source triggers for up-counting by TSG2nCNT.
5	TSG2nAT05	TSTADT0 Trigger Control 5 0: Not selected 1: Selects the matching points of TSG2nDCMP1 and TSG2nCNT as A/D source triggers for down-counting by TSG2nCNT. This bit is only valid in HT-PWM mode. In other modes, TSG2nCNT does not count down.
4	TSG2nAT04	TSTADT0 Trigger Control 4 0: Not selected 1: Selects the matching points of TSG2nDCMP1 and TSG2nCNT as A/D source triggers for up-counting by TSG2nCNT.

Table 23.10 TSG2nCTL5 Register Contents (2/2)

Bit Position	Bit Name	Function
3	TSG2nAT03	TSTADT0 Trigger Control 3 0: Not selected 1: Selects the matching points of TSG2nDCMP0 and TSG2nCNT as A/D source triggers for down-counting by TSG2nCNT. This bit is only valid in HT-PWM mode. In other modes, TSG2nCNT does not count down.
2	TSG2nAT02	TSTADT0 Trigger Control 2 0: Not selected 1: Selects the matching points of TSG2nDCMP0 and TSG2nCNT as A/D source triggers for up-counting by TSG2nCNT.
1	TSG2nAT01	TSTADT0 Trigger Control 1 0: Not selected 1: Selects INTTSG2nIPEK (*) as A/D source trigger. (*) INTTSG2nIPEK is a signal for skipping by TSG2nRCC04 to TSG2nRCC00 bits.
0	TSG2nAT00	TSTADT0 Trigger Control 0 0: Not selected 1: Selects INTTSG2nIVLY (*) as A/D source trigger. This bit is only valid in HT-PWM mode. (*) INTTSG2nIVLY is a signal for skipping by TSG2nRCC04 to TSG2nRCC00 bits

23.4.2.6 TSG2nCTL6 — TSG2n Control Register 6

This register controls A/D conversion trigger (TSTADT1) function.

See **Section 23.9, A/D Conversion Trigger Function**.

Access: This register can be read/written in 16-bit units.

Address: TSG20: FFE7 000C$_H$
TSG21: FFE7 100C$_H$

Value after reset: 0000$_H$

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	TSG2nACC [11:10]		TSG2n AT19	TSG2n AT18	TSG2n AT17	TSG2n AT16	TSG2n AT15	TSG2n AT14	TSG2n AT13	TSG2n AT12	TSG2n AT11	TSG2n AT10
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 23.11 TSG2nCTL6 Register Contents (1/2)

Bit Position	Bit Name	Function
15 to 12	Reserved	When read, the value after reset is read. When writing, write the value after reset.
11, 10	TSG2nACC [11:10]	TSTADT1 Skipping Control 00: No TSTADT1 skipping 01: 1 skipping/1 trigger output out of 2 A/D source triggers 10: 3 skipping/1 trigger output out of 4 A/D source triggers 11: 7 skipping/1 trigger output out of 8 A/D source triggers When a write access is made, the TSTADT1 skipping counter is simultaneously cleared.
9	TSG2nAT19	TSTADT1 Trigger Control 9 0:Not selected 1:Selects TSG2nSBC peak timing as A/D source trigger. This bit is only valid in HT-PWM mode.
8	TSG2nAT18	TSTADT1 Trigger Control 8 0: Not selected 1: Selects TSG2nSBC trough timing as A/D source trigger. This bit is only valid in HT-PWM mode.
7	TSG2nAT17	TSTADT1 Trigger Control 7 0: Not selected 1: Selects the matching points of TSG2nDCMP2 and TSG2nCNT as A/D source triggers for down-counting by TSG2nCNT. This bit is only valid in HT-PWM mode. In other modes, TSG2nCNT does not count down.
6	TSG2nAT16	TSTADT1 Trigger Control 6 0: Not selected 1: Selects the matching points of TSG2nDCMP2 and TSG2nCNT as A/D source triggers for up-counting by TSG2nCNT.
5	TSG2nAT15	TSTADT1 Trigger Control 5 0: Not selected 1: Selects the matching points of TSG2nDCMP1 and TSG2nCNT as A/D source triggers for down-counting by TSG2nCNT. This bit is only valid in HT-PWM mode. In other modes, TSG2nCNT does not count down.
4	TSG2nAT14	TSTADT1 Trigger Control 4 0: Not selected 1: Selects the matching points of TSG2nDCMP1 and TSG2nCNT as A/D source triggers for up-counting by TSG2nCNT.

Table 23.11 TSG2nCTL6 Register Contents (2/2)

Bit Position	Bit Name	Function
3	TSG2nAT13	TSTADT1 Trigger Control 3 0: Not selected 1: Selects the matching points of TSG2nDCMP0 and TSG2nCNT as A/D source triggers for down-counting by TSG2nCNT. This bit is only valid in HT-PWM mode. In other modes, TSG2nCNT does not count down.
2	TSG2nAT12	TSTADT1 Trigger Control 2 0: Not selected 1: Selects the matching points of TSG2nDCMP0 and TSG2nCNT as A/D source triggers for up-counting by TSG2nCNT.
1	TSG2nAT11	TSTADT1 Trigger Control 1 0: Selects INTTSG2nIPEK (*) as A/D source trigger. 1: Selects the matching points of TSG2nDCMP2 and TSG2nCNT as A/D source triggers for up-counting by TSG2nCNT. (*) INTTSG2nIPEK is a signal for skipping by TSG2nRCC04 to TSG2nRCC00 bits.
0	TSG2nAT10	TSTADT1 Trigger Control 0 0: Not selected 1: Selects INTTSG2nIVLY (*) as A/D source trigger. This bit is only valid in HT-PWM mode. (*) INTTSG2nIVLY is a signal for skipping by TSG2nRCC04 to TSG2nRCC00 bits

23.4.2.7 TSG2nIOC0 — TSG2n I/O Control Register 0

This register controls the timer output (TSOn1 to TSOn6) function and can be written only when TSG2nTE = 0 (while macros are stopped).

See **Section 23.5.4, List of Outputs in Each Mode**.

Access: This register can be read/written in 8-bit units.

Address: TSG20: FFE7 0200_H
TSG21: FFE7 1200_H

Value after reset: $7E_H$

Bit	7	6	5	4	3	2	1	0
	—	TSG2n TOE6	TSG2n TOE5	TSG2n TOE4	TSG2n TOE3	TSG2n TOE2	TSG2n TOE1	—
Value after reset	0	1	1	1	1	1	1	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R

Table 23.12 TSG2nIOC0 Register Contents

Bit Position	Bit Name	Function
7	Reserved	When read, the value after reset is read. When writing, write the value after reset.
6 to 1	TSG2nTOEm	0: The TSG2nIOC2 register (TSG2nTOm/TSG2nTOLm bits) can be rewritten by software. 1: The TSG2nIOC2 register (TSG2nTOm/TSG2nTOLm bits) cannot be rewritten by software. (Write access to the TSG2nIOC2 register is ignored.) This register controls write access to TSG2nIOC2 (m = 1, 2, 3, 4, 5, 6).
0	Reserved	When read, the value after reset is read. When writing, write the value after reset.

23.4.2.8 TSG2nIOC1 — TSG2n I/O Control Register 1

This register controls various I/O port functions.

This register can be written only when TSG2nTE = 0 (while macros are stopped).

Access: This register can be read/written in 8-bit units.

Address: TSG20: FFE7 0204_H
TSG21: FFE7 1204_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	TSG2n EOC	TSG2n WOC	TSG2n TGS	TSG2n TOS
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 23.13 TSG2nIOC1 Register Contents

Bit Position	Bit Name	Function
7 to 4	Reserved	When read, the value after reset is read. When writing, write the value after reset.
3	TSG2nEOC	INTTSG2nIER Output Control 0: Fixes the INTTSG2nIER signal to 0. 1: Enables output of the INTTSG2nIER signal.
2	TSG2nWOC	INTTSG2nIWN Output Control 0: Fixes the INTTSG2nIWN signal to 0. 1: Enables to output INTTSG2nIWN signal.
1	TSG2nTGS	TSOn7 Function Selection 0: A/D trigger output mode 1: Diagnostic output mode See **Section 23.5.4.1,(1) TSOn7 Pin Output Control**.
0	TSG2nTOS	TSOn0 Output Status Selection 0: Outputs the TSG2nCNT up/down status (same as TSG2nCUF). 1: Outputs the TSG2nSBC up/down status (same as TSG2nSUF).

23.4.2.9 TSG2nIOC2 — TSG2n I/O Control Register 2

This register controls the timer output (TSOn1 to TSOn6) functions.

See **Section 23.5.4, List of Outputs in Each Mode**.

Access: This register can be read/written in 16-bit units.

Address: TSG20: FFE7 0000_H
TSG21: FFE7 1000_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	TSG2nOL6	TSG2nOL5	TSG2nOL4	TSG2nOL3	TSG2nOL2	TSG2nOL1	—	—	TSG2nTO6	TSG2nTO5	TSG2nTO4	TSG2nTO3	TSG2nTO2	TSG2nTO1	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R

Table 23.14 TSG2nIOC2 Register Contents

Bit Position	Bit Name	Function
15	Reserved	When read, the value after reset is read. When writing, write the value after reset.
14 to 9	TSG2nOLm	TSOnm Active Level Control 0: The active level of TSOnm is high. 1: The active level of TSOnm is low. These bits control the active level of TSOnm and thereby the level selector is placed closest to the TSOnm pin. They can be changed when TSG2nTOEn = 0. In this document, TSOnm is described as TSG2nOLm = 0 (m = 1, 2, 3, 4, 5, 6).
8, 7	Reserved	When read, the value after reset is read. When writing, write the value after reset.
6 to 1	TSG2nTOm	TSOnm Output Buffer Latch 0: TSOnm is at the low level. 1: TSOnm is at the high level. These bits are TSOnm output buffer latches. They can be changed when TSG2nTOEm = 0 (m = 1, 2, 3, 4, 5, 6).
0	Reserved	When read, the value after reset is read. When writing, write the value after reset.

23.4.2.10 TSG2nIOC3 — TSG2n I/O Control Register 3

This register controls timer output (TSOn1 to TSOn6) function.

See **Section 23.5.4, List of Outputs in Each Mode**.

Access: This register can be read/written in 32-bit units.

Address: TSG20: FFE7 0074_H
TSG21: FFE7 1074_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	TSG2n TOL6	TSG2n TOL5	TSG2n TOL4	TSG2n TOL3	TSG2n TOL2	TSG2n TOL1	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R

Table 23.15 TSG2nIOC3 Register Contents

Bit Position	Bit Name	Function
31 to 7	Reserved	When read, the value after reset is read. When writing, write the value after reset.
6 to 1	TSG2nTOLm	TSOnm 0: Normal set/clear level 1: Reversed set/clear level These bits control the setting and clearing levels of the timer output. A change to an output level is reflected at the time of the next compare match. TSG2nTOLm should not be set to 1 in HT-PWM mode. In this document, TSOnm is described as TSG2nOLm = 0 (m = 1, 2, 3, 4, 5, 6)
0	Reserved	When read, the value after reset is read. When writing, write the value after reset.

23.4.2.11 TSG2nSTR0 — TSG2n Status Register 0

This register indicates TSG2 up-/down-count, reload request, and macro enable statuses.

Access: This register can be read in 8-bit units.

Address: TSG20: FFE7 0010_H
TSG21: FFE7 1010_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	TSG2nCUF	TSG2nSUF	TSG2nRSF	TSG2nTE
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 23.16 TSG2nSTR0 Register Contents

Bit Position	Bit Name	Function
7 to 4	Reserved	When read, the value after reset is read. When writing, write the value after reset.
3	TSG2nCUF	TSG2nCNT UP-/Down-Count Status Flag 0: TSG2nCNT is being incremented. 1: TSG2nCNT is being decremented.
2	TSG2nSUF	TSG2nSBC Up-/Down-Count Status Flag 0: TSG2nSBC is being incremented. 1: TSG2nSBC is being decremented. This bit is only valid in HT-PWM mode. (In other mode, it is fixed to 0.)
1	TSG2nRSF	Reload Holding Status Flag 0: No reload request, or reloading is complete. 1: A reload request is generated and waiting for the reload timing. See **Section 23.7.3, Reload Request Flag (TSG2nRSF)**.
0	TSG2nTE	Timer Enable Status Flag 0: The timer is stopped. (Macros are stopped.) 1: The timer is operating.

23.4.2.12 TSG2nSTR1 — TSG2n Status Register 1

This register indicates the direction, reload request and macro enable statuses of TAPTSn2-0.

Access: This register can be read in 8-bit units.

Address: TSG20: FFE7 0014_H
TSG21: FFE7 1014_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	TSG2n TSF	TSG2nOPF[2:0]		
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 23.17 TSG2nSTR1 Register Contents

Bit Position	Bit Name	Function
7 to 4	Reserved	When read, the value after reset is read. When writing, write the value after reset.
3	TSG2nTSF	TAPTSn2-0 Pattern Order Detection Flag 0: Indicates that the rotation order of TAPTSn2-0 is clockwise. 1: Indicates that the rotation order of TAPTSn2-0 is counter-clockwise. See **Section 23.7.5, Pattern Order Detection Flag (TSG2nTSF)** and **Section 23.7.7, Pattern Inversion Detection Flag (TSG2nPRF)**.
2 to 0	TSG2nOPF[2:0]	Timer Output Pattern Flag 000-111: Indicates the timer output pattern status.

23.4.2.13 TSG2nSTR2 — TSG2n Status Register 2

This register indicates the error/warning status.

Access: This register can be read in 16-bit units.

Address: TSG20: FFE7 0018_H
TSG21: FFE7 1018_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TSG2n TBF2	TSG2n TBF1	TSG2n TBF0	TSG2n PPF	TSG2n PEF	—	TSG2n NDF	TSG2n PRF	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 23.18 TSG2nSTR2 Register Contents (1/2)

Bit Position	Bit Name	Function
15 to 10	Reserved	When read, the value after reset is read. When writing, write the value after reset.
9	TSG2nTBF2	Positive/Negative Phase Simultaneous Activation Detection Flag 2 0: No error 1: TSOn5 and TSOn6 are active simultaneously or the setting of the compare register may be incorrect For how to clear this error flag, see the description of the TSG2nTBF0 bit.
8	TSG2nTBF1	Positive/Negative Phase Simultaneous Activation Detection Flag 1 0: No error 1: TSOn3 and TSOn4 are active simultaneously or the setting of the compare register may be incorrect. For how to clear this error flag, see the description of the TSG2nTBF0 bit.
7	TSG2nTBF0	Positive/Negative Phase Simultaneous Activation Detection Flag 0 0: No error 1: TSOn1 and TSOn2 are active simultaneously or the setting of the compare register may be incorrect. This error flag can be cleared in the following two ways. • Write 1 to the TSG2nTBR2-0 bits. • Change TSG2nTE from 0 to 1. See **Section 23.4.2.14, TSG2nSTC — TSG2n Status Clear Trigger Registers**.
6	TSG2nPPF	Input/Output Pattern Phase Difference Detection Flag 0: No error 1: A phase difference detected between the input patterns (TAPTSn2-0) and the output patterns (TSG2nOPF2-0) This warning flag can be cleared in the following two ways. • Write 1 to the TSG2nPPR bit. • Write 1 to TSG2nTS or TSTSST input (timer start or timer restart). See **Section 23.7.8, Pattern Phase Difference Detection Flag (TSG2nPPF)**.
5	TSG2nPEF	Input Pattern (TAPTSn2-0) Error Detection Flag 0: No error 1: Detects that TAPTSn2-0 are all high or low. This warning flag can be cleared in the following two ways. • Write 1 to TSG2nPER bit. • Write 1 to TSG2nTS or TSTSST input (timer start or timer restart). See **Section 23.7.6, Pattern Error Detection Flag (TSG2nPEF)**.
4	Reserved	When read, the value after reset is read. When writing, write the value after reset.

Table 23.18 TSG2nSTR2 Register Contents (2/2)

Bit Position	Bit Name	Function
3	TSG2nNDF	Input Pattern (TAPTSn2-0) Noise Detection Flag 0: No error 1: Detects non-gray code operation of TAPTSn2-0 (the level of TAPTSn2-0 on two or more pins changes simultaneously). This warning flag can be cleared in the following two ways. • Write 1 to the TSG2nNDR bit. • Write 1 to TSG2nTS or TSTSST input (timer start or timer restart). See **Section 23.7.4, Noise Detection Flag (TSG2nNDF)**.
2	TSG2nPRF	Input Pattern (TAPTSn2) Inversion Detection Flag 0: No error 1: Detects inversion of TAPTSn2-0 (detects the change of motor rotation.) This warning flag can be cleared in the following two ways. • Write 1 to the TSG2nPRR bit. • Write 1 to TSG2nTS or TSTSST input (timer start or timer restart). See **Section 23.7.5, Pattern Order Detection Flag (TSG2nTSF)**.
1, 0	Reserved	When read, the value after reset is read. When writing, write the value after reset.

23.4.2.14 TSG2nSTC — TSG2n Status Clear Trigger Registers

This register is a trigger register for clearing TSG2nSTR2.

When this register is read, it is always read as "0000_H".

Access: This register can be written in 16-bit units.

Address: TSG20: FFE7 001C$_H$
TSG21: FFE7 101C$_H$

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TSG2nTBR2	TSG2nTBR1	TSG2nTBR0	TSG2nPPR	TSG2nPER	—	TSG2nNDR	TSG2nPRR	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	W	W	W	W	W	R	W	W	R	R

Table 23.19 TSG2nSTC Register Contents (1/2)

Bit Position	Bit Name	Function
15 to 10	Reserved	When read, the value after reset is read. When writing, write the value after reset.
9	TSG2nTBR2	TSG2nTBF2 Clear Trigger 0 write: Invalid 1 write: Clears TSG2nTBF2 to 0. If TSG2nTBF2 is set and 1 is written to TSG2nTBR2 at the same time, the TSG2nTBF2 setting is given priority.
8	TSG2nTBR1	TSG2nTBF1 Clear Trigger 0 write: Invalid 1 write: Clears TSG2nTBF1 to 0. If TSG2nTBF1 is set and 1 is written to TSG2nTBR1 at the same time, the TSG2nTBF1 setting is given priority.
7	TSG2nTBR0	TSG2nTBF0 Clear Trigger 0 write: Invalid 1 write: Clears TSG2nTBF0 to 0. If TSG2nTBF0 is set and 1 is written to TSG2nTBR0 at the same time, the TSG2nTBF0 setting is given priority.
6	TSG2nPPR	TSG2nPPF Clear Trigger 0 write: Invalid 1 write: Clears TSG2nPPF to 0 If TSG2nPPF is set and 1 is written to TSG2nPPR at the same time, the TSG2nPPF setting is given priority.
5	TSG2nPER	TSG2nPEF Clear Trigger 0 write: Invalid 1 write: Clears TSG2nPEF to 0 If TSG2nPEF is set and 1 is written to TSG2nPER at the same time, the TSG2nPEF setting is given priority.
4	Reserved	When read, the value after reset is read. When writing, write the value after reset.
3	TSG2nNDR	TSG2nNDF Clear Trigger 0 write: Invalid 1 write: Clears TSG2nNDF to 0 If TSG2nNDF is set and 1 is written to TSG2nNDR at the same time, the TSG2nNDF setting is given priority.
2	TSG2nPRR	TSG2nPRF Clear Trigger 0 write: Invalid 1 write: Clears TSG2nPRF to 0 If TSG2nPRF is set and 1 is written to TSG2nPRR at the same time, the TSG2nPRF setting is given priority.

Table 23.19 TSG2nSTC Register Contents (2/2)

Bit Position	Bit Name	Function
1, 0	Reserved	When read, the value after reset is read. When writing, write the value after reset.

23.4.2.15 TSG2nOPT0 — TSG2n Option Register 0

This register configures the option function.

Access: This register can be read/written in 8-bit units.

Address: TSG20: FFE7 0020_H
TSG21: FFE7 1020_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	TSG2n SOC	TSG2n STE	—	TSG2n PSS	TSG2n IDC	TSG2n PSC	—
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R	R/W	R/W	R/W	R

Table 23.20 TSG2nOPT0 Register Contents

Bit Position	Bit Name	Function
7	Reserved	When read, the value after reset is read. When writing, write the value after reset.
6	TSG2nSOC	Enables or disables control of the timer output (TSOn1 to TSOn6 pins) by software. 0 : Disables control by software. 1 : Enables control by software. • When TSG2nSOC is set to 1, timer output is switched to the software control/trigger control output pattern specified by TSG2nSPC2 to TSG2nSPC0. The dead time is secured by the dead time counter.
5	TSG2nSTE	Enables or disables control by the pattern output trigger. 0: Disables TAPTSn0 to TAPTSn2 inputs. 1: Enables TAPTSn0 to TAPTSn2 inputs. • TSG2nSTE is valid in 120-DC mode and while software output control function is enabled.
4	Reserved	When read, the value after reset is read. When writing, write the value after reset.
3	TSG2nPSS	Selects the pattern output order switch factor. 0: Switching of the pattern output order by TSG2nPSC is not used. 1: Switching of the pattern output order by TSG2nPSC is used.
2	TSG2nIDC	Determines the output pattern from the TSOn1 to TSOn6 pins in combination with the TSG2nIDC, TSG2nSTR1.TSG2nTSF and TSG2nPSC signals. For the timer output order and patterns to be output, see **Figure 23.74** to **Figure 23.77**, Example of Operation in 120-DC Mode, in **Section 23.11.4.5, Operation in 120-DC Mode**.
1	TSG2nPSC	Selects the pattern output order when the semi-automatic cruise function is enabled. 0: Switches the timer output (TSOn1 to TSOn6) in the normal rotation. 1: Switches the timer output (TSOn1 to TSOn6) in the reverse rotation. • If the signal input to TAPTSn0 to TAPTSn2 changes with TSG2n operation being stopped (TSG2nSTR0.TSG2nTE = 0), the TSG2nTRG0.TSG2nTS bit should be set to 1 after matching the input signal change logic with the TSG2nPSC order. • For output order in normal or reverse rotation, see **Section 23.11.4, 120-DC Mode**. Here, normal rotation and reverse rotation refer to the change of output, and they are different from normal rotation and reverse rotation of a motor.
0	Reserved	When read, the value after reset is read. When writing, write the value after reset.

23.4.2.16 TSG2nOPT1 — TSG2n Option Register 1

This register configures the option function.

Access: This register can be read/written in 8-bit units.

Address: TSG20: FFE7 0024_H
TSG21: FFE7 1024_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	TSG2nSPC[2:0]		
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W

Table 23.21 TSG2nOPT1 Register Contents

Bit Position	Bit Name	Function
7 to 3	Reserved	When read, the value after reset is read. When writing, write the value after reset.
2 to 0	TSG2nSPC[2:0]	These bits specify the timer output pattern while software output function is enabled and in 120-DC mode. For the output pattern, see **Section 23.11.5, Software Output Control Function**, and **Section 23.11.4, 120-DC Mode**.

23.4.2.17 TSG2nTRG0 — TSG2n Trigger Register 0

This register is a trigger register for starting the timer.

When this register is read, it is always read as "00_H".

Access: This register can be written in 8-bit units.

Address: TSG20: FFE7 0030_H
TSG21: FFE7 1030_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TSG2nTS
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 23.22 TSG2nTRG0 Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When read, the value after reset is read. When writing, write the value after reset.
0	TSG2nTS	Timer Start Trigger 0: Invalid 1: Triggers the start of timer software (setting TSG2nTE = 1) Also, this bit triggers the restart of timer software.

23.4.2.18 TSG2nTRG1 — TSG2n Trigger Register 1

This register controls the stop of the timer.

When this register is read, it is always read as "00_H".

Access: This register can be written in 8-bit units.

Address: TSG20: FFE7 0034_H
TSG21: FFE7 1034_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TSG2nTT
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 23.23 TSG2nTRG1 Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When read, the value after reset is read. When writing, write the value after reset.
0	TSG2nTT	Timer Stop Trigger 0: Invalid 1: Triggers stopping of timer software (TSG2nTE = 0).

23.4.2.19 TSG2nCNT — TSG2n Counter Register

These registers are the main counters for this macro. In HT-PWM mode, TSG2nCNT counts up or down according to triangle-wave control where the counter value is incremented by two until the peak and then decremented by two until the trough. In other modes, the counter is under sawtooth wave control where its value is incremented by one until the peak and then returned to the initial value.

Access: This register can be read in 16-bit units.

Address: TSG20: FFE7 0028_H
TSG21: FFE7 1028_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	16-bit counter															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

[TSG2nCNT Count Value]

Operating Mode	At the Beginning	Minimum Value	Maximum Value
HT-PWM mode	TSG2nDTC0	TSG2nDTC0	TSG2nDTC0+TSG2nCMP0*1
Other modes	0000_H	0000_H	TSG2nCMP0

Note 1. Set a value so that the following condition can be satisfied: TSG2nDTC0 + TSG2nCMP0 < $FFFF_H$.

23.4.2.20 TSG2nSBC — TSG2n Sub-Counter Register

This register is a sub-counter of this macro in HT-PWM mode. In HT-PWM mode, TSG2nCNT counts up or down according to triangle-wave control where the counter value is incremented by two until the peak and then decremented by two until the trough. In addition, this register is provided with a jump function.

Access: This register can be read in 16-bit units.

Address: TSG20: FFE7 002C$_H$
TSG21: FFE7 102C$_H$

Value after reset: 0000$_H$

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	16-bit counter															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Operating Mode	At the Beginning	Minimum Value	Maximum Value
HT-PWM mode	TSG2nDTC0	0000$_H$	TSG2nDTC0+TSG2nDTC1+TSG2nCMP0*1
Other modes	0000$_H$	0000$_H$	0000$_H$

Note 1. Set a value so that the following condition can be satisfied: TSG2nDTC0 + TSG2nDTC1 + TSG2nCMP0 < FFFF$_H$.

23.4.2.21 TSG2nCMP0 — TSG2n Compare Register 0

This register is a compare register that specifies the PWM cycle in all modes.

Access: This register can be read/written in 32-bit units.

Address: TSG20: FFE7 0058_H
TSG21: FFE7 1058_H

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	16-bit compare register															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Operating Mode	At the Beginning	Minimum Value	Maximum Value
HT-PWM mode	TSG2nCMP0*[1]	0002_H	$FFFE_H$
Other modes	TSG2nCMP0 + 1	1 (TSG2nCMP0 = 0000_H)	10000_H (TSG2nCMP0 = $FFFF_H$)

Note 1. In HT-PWM mode, the LSB is ignored because TSG2nCNT is incremented or decremented by 2.

23.4.2.22 TSG2nCMP1W — TSG2n Compare Registers 1 and 2

These registers specify the timer PWM duty in all modes. The 16 high-order bits are called TSG2nCMP2 and the 16 low-order bits are called TSG2nCMP1.

The table below shows the matching operation of TSG2nCNT and the compare registers.

Access: This register can be read/written in 32-bit units.

Address: TSG20: FFE7 0040_H
TSG21: FFE7 1040_H

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TSG2nCMP2 (16-bit compare register)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSG2nCMP1 (16-bit compare register)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

	PWM Mode	HT-PWM Mode	SP-PWM Mode	120-DC Mode
TSG2nCMP1	TSOn1 clear	TSOn1 clear/ TSOn2 set	Same as on the left	TSOn1, 3, or 5 output PWM is selected by the TSG2nPAT0 register
TSG2nCMP2	TSOn1 set	TSOn1 set/ TSOn2 clear	Same as on the left	TSOn1, 3, or 5 output PWM is selected by the TSG2nPAT0 register

When TSG2nTOEm = 1, the dead time function is enabled in any operating mode. The table above shows the basic functions.

In HT-PWM mode, this is connected with the timing of matches between this register and TSG2nSBC.

In 120-DC mode, the output from TSOn1 to TSOn6 is controlled by TSG2nCMPm, TSG2nPAT0 and TSG2nPAT1.

23.4.2.23 TSG2nCMP5W — TSG2n Compare Registers 5 and 6

These registers specify the timer PWM duty in all modes. The 16 high-order bits are called TSG2nCMP6 and the 16 low-order bits are called TSG2nCMP5.

The table below shows the matching operation of TSG2nCNT and the compare registers.

Access: These registers can be read/written in 32-bit units.

Address: TSG20: FFE7 0044_H
TSG21: FFE7 1044_H

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TSG2nCMP6 (16-bit compare register)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSG2nCMP5 (16-bit compare register)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

	PWM Mode	HT-PWM Mode	SP-PWM Mode	120-DC Mode
TSG2nCMP5	TSOn3 clear	TSOn3 clear/ TSOn4 set	Same as on the left	TSOn1, 3, or 5 output PWM is selected by the TSG2nPAT0 register
TSG2nCMP6	TSOn3 set	TSOn3 clear/ TSOn4 set	Same as on the left	TSOn1, 3, or 5 output PWM is selected by the TSG2nPAT0 register

When TSG2nTOEm = 1, the dead time function is enabled in any operating mode. The table above shows the basic functions.

In HT-PWM mode, this is connected with the timing of matches between this register and TSG2nSBC.

In 120-DC mode, the output from TSOn1 to TSOn6 is controlled by TSG2nCMPm, TSG2nPAT0 and TSG2nPAT1.

23.4.2.24 TSG2nCMP9W — TSG2n Compare Registers 9 and 10

These registers specify the timer PWM duty in all modes. The 16 high-order bits are called TSG2nCMP10 and the 16 low-order bits are called TSG2nCMP9.

The table below shows the matching operation of TSG2nCNT and the compare registers.

Access: These registers can be read/written in 32-bit units.

Address: TSG20: FFE7 0048_H
TSG21: FFE7 1048_H

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TSG2nCMP10 (16-bit compare register)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSG2nCMP9 (16-bit compare register)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

	PWM Mode	HT-PWM Mode	SP-PWM Mode	120-DC Mode
TSG2nCMP9	TSOn5 clear	TSOn5 clear/ TSOn6 set	Same as on the left	TSOn1, 3, or 5 output PWM is selected by the TSG2nPAT0 register
TSG2nCMP10	TSOn5 set	TSOn5 set/ TSOn6 clear	Same as on the left	TSOn1, 3, or 5 output PWM is selected by the TSG2nPAT0 register

When TSG2nTOEm = 1, the dead time function is enabled in any operating mode. The table above shows the basic functions.

In HT-PWM mode, this is connected with the timing of matches between this register and TSG2nSBC.

In 120-DC mode, the output from TSOn1 to TSOn6 is controlled by TSG2nCMPm, TSG2nPAT0 and TSG2nPAT1.

23.4.2.25 TSG2nCMP3W — TSG2n Compare Registers 3 and 4

These registers specify the timer PWM duty in all modes. The 16 high-order bits are called TSG2nCMP4 and the 16 low-order bits are called TSG2nCMP3.

The table below shows the matching operation of TSG2nCNT and the compare registers.

Access: These registers can be read/written in 32-bit units.

Address: TSG20: FFE7 004C$_H$
TSG21: FFE7 104C$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TSG2nCMP4 (16-bit compare register)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSG2nCMP3 (16-bit compare register)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

	PWM Mode	HT-PWM Mode	SP-PWM Mode	120-DC Mode
TSG2nCMP3	TSOn2 clear	—	—	TSOn2, 4, or 6 output PWM is selected by the TSG2nPAT1 register
TSG2nCMP4	TSOn2 set	—	—	TSOn2, 4, or 6 output PWM is selected by the TSG2nPAT1 register

When TSG2nTOEm = 1, the dead time function is enabled in any operating mode. The table above shows the basic functions.

In HT-PWM mode, this is connected with the timing of matches between this register and TSG2nSBC.

In 120-DC mode, the output from TSOn1 to TSOn6 is controlled by TSG2nCMPm, TSG2nPAT0 and TSG2nPAT1.

23.4.2.26 TSG2nCMP7W — TSG2n Compare Registers 7 and 8

These registers specify the timer PWM duty in any mode. The 16 high-order bits are called TSG2nCMP8 and the 16 low-order bits are called TSG2nCMP7.

The table below shows the matching operation of TSG2nCNT and the compare registers.

Access: These registers can be read/written in 32-bit units.

Address: TSG20: FFE7 0050_H
TSG21: FFE7 1050_H

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TSG2nCMP8 (16-bit compare register)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSG2nCMP7 (16-bit compare register)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

	PWM Mode	HT-PWM Mode	SP-PWM Mode	120-DC Mode
TSG2nCMP7	TSOn4 clear	—	—	TSOn2, 4, or 6 output PWM is selected by the TSG2nPAT1 register
TSG2nCMP8	TSOn4 set	—	—	TSOn2, 4, or 6 output PWM is selected by the TSG2nPAT1 register

When TSG2nTOEm = 1, the dead time function is enabled in any operating mode. The table above shows the basic functions.

In HT-PWM mode, this is connected with the timing of matches between this register and TSG2nSBC.

In 120-DC mode, the output from TSOn1 to TSOn6 is controlled by TSG2nCMPm, TSG2nPAT0 and TSG2nPAT1.

23.4.2.27 TSG2nCMP11W — TSG2n Compare Registers 11 and 12

These registers specify the timer PWM duty in any mode. The 16 high-order bits are called TSG2nCMP12 and the 16 low-order bits are called TSG2nCMP11.

The table below shows the matching operation of TSG2nCNT and the compare registers.

Access: These registers can be read/written in 32-bit units.

Address: TSG20: FFE7 0054_H
TSG21: FFE7 1054_H

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TSG2nCMP12 (16-bit compare register)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSG2nCMP11 (16-bit compare register)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

	PWM Mode	HT-PWM Mode	SP-PWM Mode	120-DC Mode
TSG2nCMP11	TSOn6 clear	—	—	TSOn2, 4, or 6 output PWM is selected by the TSG2nPAT1 register
TSG2nCMP12	TSOn6 set	—	—	TSOn2, 4, or 6 output PWM is selected by the TSG2nPAT1 register

When TSG2nTOEm = 1, the dead time function is enabled in any operating mode. The table above shows the basic functions.

In HT-PWM mode, this is connected with the timing of matches between this register and TSG2nSBC.

In 120-DC mode, the output from TSOn1 to TSOn6 is controlled by TSG2nCMPm, TSG2nPAT0 and TSG2nPAT1.

23.4.2.28 TSG2nCMP1-12 — TSG2n Compare Registers 1 to 12

These registers specify the timer PWM duty in all modes. They reflect the address mapping of TSG2nCMP1W, 3W, 5W, 7W, 9W, and 11W, so the function is the same as that of these registers.

Access: These registers can be read/written in 16-bit units.

Address:
TSG20CMP1/TSG21CMP1: FFE7 0080_H/FFE7 1080_H
TSG20CMP2/TSG21CMP2: FFE7 0084_H/FFE7 1084_H
TSG20CMP3/TSG21CMP3: FFE7 0098_H/FFE7 1098_H
TSG20CMP4/TSG21CMP4: FFE7 $009C_H$/FFE7 $109C_H$
TSG20CMP5/TSG21CMP5: FFE7 0088_H/FFE7 1088_H
TSG20CMP6/TSG21CMP6: FFE7$008C_H$/FFE7 $108C_H$
TSG20CMP7/TSG21CMP7: FFE7 $00A0_H$/FFE7 $10A0_H$
TSG20CMP8/TSG21CMP8: FFE7 $00A4_H$/FFE7 $10A4_H$
TSG20CMP9/TSG21CMP9: FFE7 0090_H/FFE7 1090_H
TSG20CMP10/TSG21CMP10: FFE7 0094_H/FFE 71094$_H$
TSG20CMP11/TSG21CMP11: FF87 $00A8_H$/FF87 $10A8_H$
TSG20CMP12/TSG21CMP12: FF87 $00AC_H$/FF87 $10AC_H$

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSG2nCMPm (16-bit compare register)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note 1. m = 1 to 12

23.4.2.29 TSG2nDCMP0W — TSG2n Diagnostic Output Compare Registers 0 and 1

These registers diagnose the output function and A/D start trigger timing in all modes. The 16 high-order bits are called TSG2nDCMP1 and the 16 low-order bits are called TSG2nDCMP0. When the value of TSG2nCNT and these registers match, the TSTADT0, TSTADT1, and TSOn7 output pins are controlled.

See **Section 23.9, A/D Conversion Trigger Function** and **Section 23.5.4.1 (1) TSOn7 Pin Output Control**.

Access: These registers can be read/written in 32-bit units.

Address: TSG20: FFE7 005C$_H$
TSG21: FFE7 105C$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TSG2nDCMP1 (16-bit compare register)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSG2nDCMP0 (16-bit compare register)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

23.4.2.30 TSG2nDCMP2 — TSG2n Diagnostic Output Compare Register 2

This register diagnose the output function and A/D start trigger timing in all modes. The 16 low-order bits are called TSG2nDCMP2.

When the value of TSG2nCNT and this register match, the TSTADT0, TSTADT1 and TSOn7 output pins are controlled.

See **Section 23.9, A/D Conversion Trigger Function** and **Section 23.5.4.1 (1) TSOn7 Pin Output Control**.

Access: This register can be read/written in 32-bit units.

Address: TSG20: FFE7 0060_H
TSG21: FFE7 1060_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSG2nDCMP2 (16-bit compare register)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

23.4.2.31 TSG2nPAT0W — TSG2n Pattern Register 0

This register controls the UT/VT/WT output pattern.

Bits 17 to 15 are for PAT5T; bits 14 to 12 for PAT4T; bits 11 to 9 for PAT3T; bits 8 to 6 for PAT2T; bits 5 to 3 for PAT1T; and bits 2 to 0 for PAT0T.

Access: This register can be read/written in 32-bit units.

Address: TSG20: FFE7 0064_H
TSG21: FFE7 1064_H

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PAT5T	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PAT5T	PRT4T			PAT3T			PAT2T			PAT1T			PAT0T		
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Value of PAT5T to PAT0T	Output Control
000	The low level is held.
001	PWM output set by TSG2nCMP1
010	PWM output set by TSG2nCMP2
011	PWM output set by TSG2nCMP5
100	PWM output set by TSG2nCMP6
101	PWM output set by TSG2nCMP9
110	PWM output set by TSG2nCMP10
111	The high level is held.

23.4.2.32 TSG2nPAT1W — TSG2n Pattern Register 1

This register controls the UT/VT/WT output pattern.

Bits 17 to 15 are for PAT5B; bits 14 to 12 for PAT4B; bits 11 to 9 for PAT3B; bits 8 to 6 for PAT2B; bits 5 to 3 for PAT1B; and bits 2 to 0 for PAT0B.

Access: This register can be read/written in 32-bit units.

Address: TSG20: FFE7 0068_H
TSG21: FFE7 1068_H

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PAT5B	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PAT5B	PAT4B			PAT3B			PAT2B			PAT1B			PAT0B		
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Value of PAT5B to PAT0B	Output Control
000	The low level is held.
001	PWM output set by TSG2nCMP3
010	PWM output set by TSG2nCMP4
011	PWM output set by TSG2nCMP7
100	PWM output set by TSG2nCMP8
101	PWM output set by TSG2nCMP11
110	PWM output set by TSG2nCMP12
111	The high level is held.

23.4.2.33 TSG2nDTC0W — TSG2n Dead Time Compare Register 0

This register is a compare register for the dead time. It controls the period from inverse phase inactivation to positive phase activation. The 10 low-order bits are called TSG2nDTC0.

When the 10-bit dead time compare value is changed, the write protection code must be written to TSG2nPWDATA0. Only when TSG2nPWDATA0 and TSG2nDTPR (TSG2n dead time protection register) match while TSG2nDTCM = 0, the 10-bit dead time compare value can be changed.

During timer operation (TSG2nTE = 1), the register should not be written except by reloading (TSG2nRMC = 0).

Access: This register can be read/written in 32-bit units.

Address: TSG20: FFE7 006C$_H$
TSG21: FFE7 106C$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	TSG2nPWDATA0 (Write protection code)														
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TSG2nDTC0 (10-bit dead time compare)									
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

23.4.2.34 TSG2nDTC1W — TSG2n Dead Time Compare Register 1

This register is a compare register for the dead time. It controls the period from positive phase inactivation to inverse phase activation. The 10 low-order bits are called TSG2nDTC1.

When the 10-bit dead time compare value is changed, the write protection code must be written to TSG2nPWDATA1. Only when TSG2nPWDATA1 and TSG2nDTPR (TSG2n dead time protection register) match while TSG2nDTCM = 0, the 10-bit dead time compare value can be changed.

During timer operation (TSG2nTE = 1), the register should not be written except by reloading (TSG2nRMC = 0).

Access: This register can be read/written in 32-bit units.

Address: TSG20: FFE7 0070_H
TSG21: FFE7 1070_H

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	TSG2nPWDATA1 (Write protection code)														
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	TSG2nDTC1 (10-bit dead time compare)									
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

23.4.2.35 TSG2nCMPU — TSG2n U Phase Compare Register

This register is used only in HT-PWM mode. Though the function of this register is the same as TSG2nCMP1W (TSG2nCMP1, 2), access to this register requires a specific sequence. In other words, data are written to TSG2nCMPU and then stored in the TSG2nCMP1 and 2 registers. Thereby, only a single write access is required to start the generation of a symmetric triangle wave in PWM mode (see **Figure 23.2**).

When this register is read, the same value as TSG2nCMP1 is read.

Access: This register can be read/written in 16-bit units.

Address: TSG20: FFE7 00B0$_H$
TSG21: FFE7 10B0$_H$

Value after reset: 0000$_H$

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSG2nCMPU (16-bit compare register)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

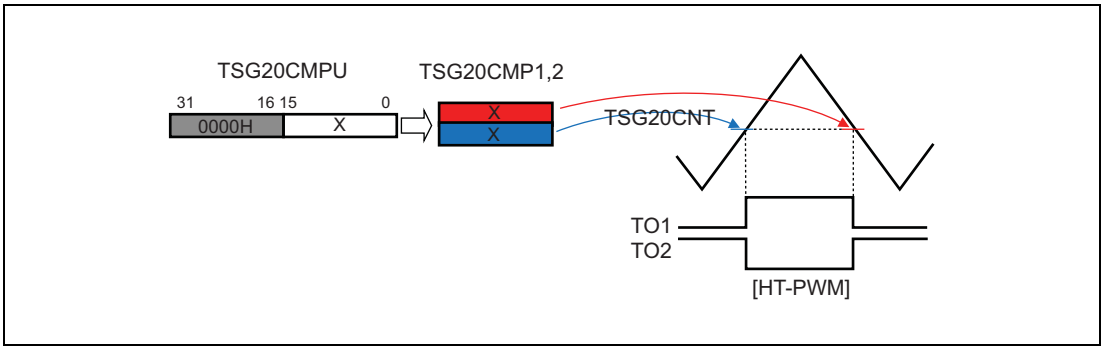


Figure 23.2 TSG2nCMPU Register Access

23.4.2.36 TSG2nCMPV — TSG2n V Phase Compare Register

This register is used only in HT-PWM mode. Though the function of this register is the same as TSG2nCMP5W (TSG2nCMP5, 6), access to this register requires a specific sequence. In other words, data are written to TSG2nCMPV and then stored in the TSG2nCMP 5 and 6 registers. Thereby, only a single write access is required to start the generation of a symmetric triangle wave in PWM mode (see **Figure 23.3**).

When this register is read, the same value as TSG2nCMP5 is read.

Access: This register can be read/written in 16-bit units.

Address: TSG20: FFE7 00B4$_H$
TSG21: FFE7 10B4$_H$

Value after reset: 0000$_H$

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSG2nCMPV (16-bit compare register)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

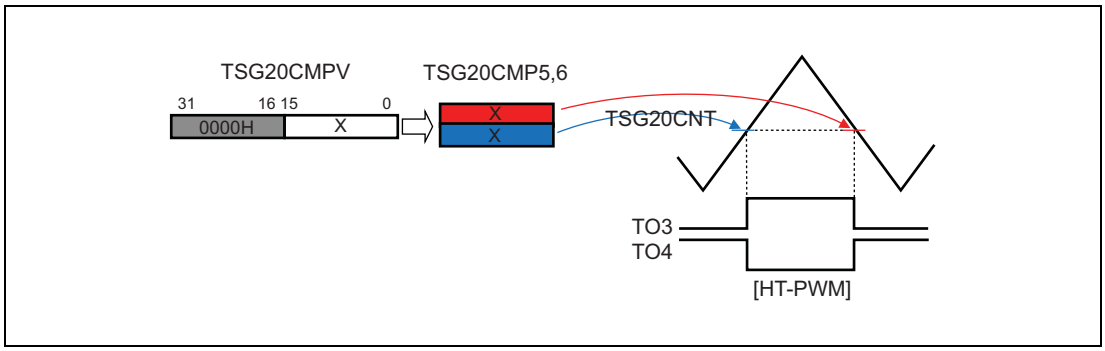


Figure 23.3 TSG2nCMPV Register Access

23.4.2.37 TSG2nCMPW — TSG2n W Phase Compare Register

This register is used only in HT-PWM mode. Though the function of this register is the same as TSG2nCMP9W (TSG2nCMP9,10), access to this register requires a specific sequence. In other words, data are written to TSG2nCMPW and then stored in the TSG2nCMP 9 and 10 registers. Thereby, only a single write access is required to start the generation of a symmetric triangle wave in PWM mode (see **Figure 23.4**).

When this register is read, the same value as TSG2nCMP9 is read.

Access: This register can be read/written in 16-bit units.

Address: TSG20: FFE7 00B8$_H$
TSG21: FFE7 10B8$_H$

Value after reset: 0000$_H$

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSG2nCMPW (16-bit compare register)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

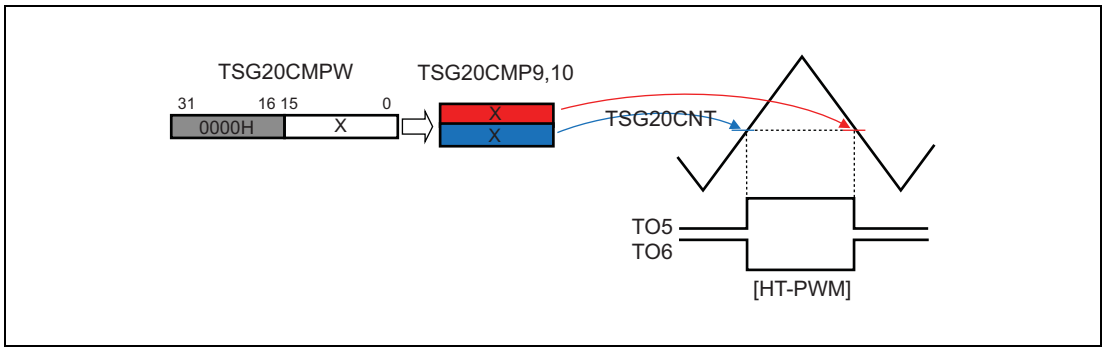


Figure 23.4 TSG2nCMPW Register Access

23.4.2.38 TSG2nUPW — TSG2n U Phase Period Register

This register is used only in SP-PWM mode and sets the PWM period. In other words, data written to TSG2nUPW and TSG2nCMP2 are stored together in the TSG2nCMP1 register (see **Figure 23.5**). When this register is read, the same value as TSG2nCMP1 is read.

Access: This register can be read/written in 16-bit units.

Address: TSG20: FFE7 00BC$_H$
TSG21: FFE7 10BC$_H$

Value after reset: 0000$_H$

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSG2nUPW (16-bit compare register)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

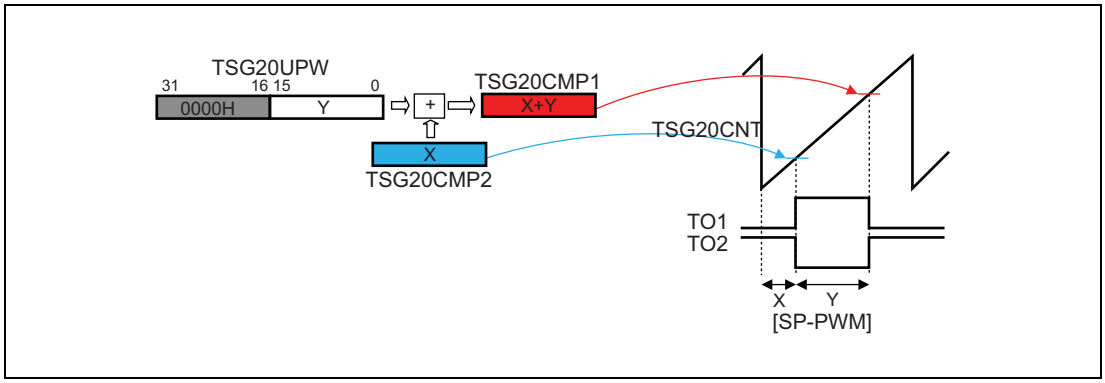


Figure 23.5 TSG2nUPW Register Access

23.4.2.39 TSG2nVPW — TSG2n V Phase Period Register

This register is used only in SP-PWM mode and sets the PWM period. In other words, data written to TSG2nVPW and TSG2nCMP6 are stored together in the TSG2nCMP5 register (see **Figure 23.6**). When this register is read, the same value as TSG2nCMP5 is read.

Access: This register can be read/written in 16-bit units.

Address: TSG20: FFE7 00C0$_H$
TSG21: FFE7 10C0$_H$

Value after reset: 0000$_H$

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSG2nVPW (16-bit compare register)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

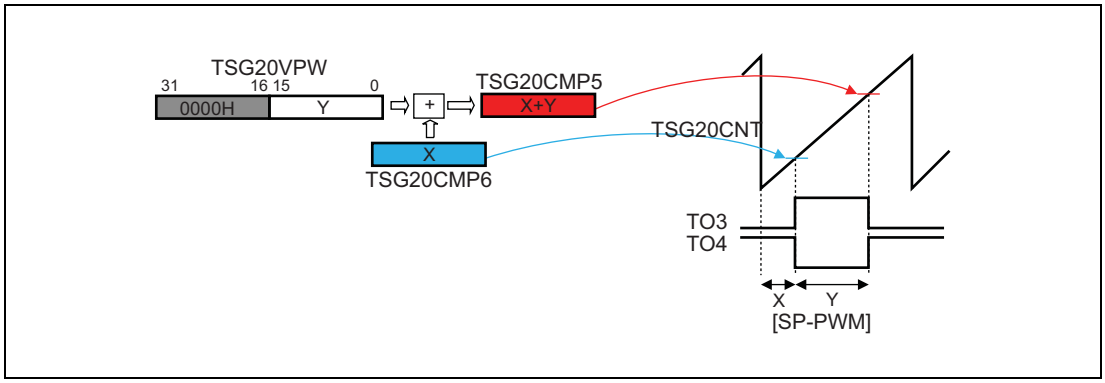


Figure 23.6 TSG2nVPW Register Access

23.4.2.40 TSG2nWPW — TSG2n W Phase Period Register

This register is used only in SP-PWM mode and sets the PWM period. In other words, data written to TSG2nWPW and TSG2nCMP10 are stored together in the TSG2nCMP9 register (see **Figure 23.7**). When this register is read, the same value as TSG2nCMP9 is read.

Access: This register can be read/written in 16-bit units.

Address: TSG20: FFE7 00C4$_H$
TSG21: FFE7 10C4$_H$

Value after reset: 0000$_H$

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSG2nWPW (16-bit compare register)															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

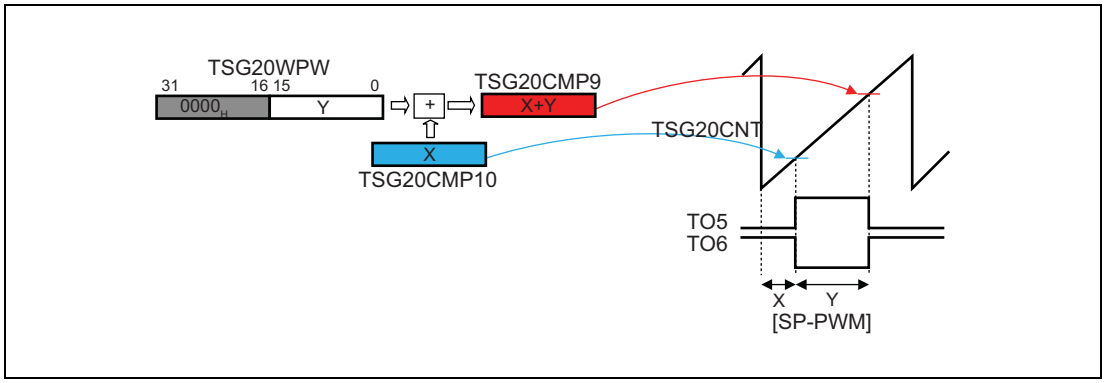


Figure 23.7 TSG2nWPW Register Access

23.4.2.41 TSG2nDTPR — TSG2n Dead Time Protection Register

This register controls protection against write access to the TSG2nDTC0 and TSG2nDTC1 registers.

Access: This register can be read/written in 16-bit units.

Address: TSG20: FFE7 0210_H
TSG21: FFE7 1210_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TSG2nDTCM	TSG2nDTPR														
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 23.24 TSG2nDTPR Register Contents

Bit Position	Bit Name	Function
15	TSG2nDTCM	TSG2nDTC0 and 1 Write Mask 0: Enables rewriting of TSG2nDTC0 and TSG2nDTC1. 1: Disables rewriting of TSG2nDTC0 and TSG2nDTC1.
14 to 0	TSG2nDTPR	Write Protection Code 0000_H-$7FFF_H$: Sets any value from 0000_H to $7FFF_H$.

23.5 Basic Operation

23.5.1 Basic Operation of 16-Bit Counter

The basic operation of the 16-bit counter is described.

Counting start

The 16-bit counter of TSG2n starts counting from the value after reset 0000_H in any mode other than HT-PWM mode.

The counter increments from 0000_H, 0001_H, 0002_H, 0003_H, … in any mode other than HT-PWM mode.

Counter clear

The 16-bit counter is cleared when the counter value and the compare register set value match.

Counter read during counting

In the TSG2n, the 16-bit counter value during counting can be read through TSG2nCNT.

Interrupt operation

In the TSG2n, the following interrupts are generated.

- INTTSG2nI[0]: This interrupt works as a cyclic interrupt when the 16-bit counter value matches the TSG2nDTC0 value in HT-PWM mode. It works as an interrupt on compare match between the 16-bit counter value and the TSG2nCMP0 buffer register value in modes other than HT-PWM mode.
- INTTSG2nI[1]: An interrupt on compare match between the 16-bit counter value and the TSG2nCMP1 buffer register.
- INTTSG2nI[2]: An interrupt on compare match between the 16-bit counter value and the TSG2nCMP2 buffer register value.
- INTTSG2nI[3]: An interrupt on compare match between the 16-bit counter value and the TSG2nCMP3 buffer register value.
- INTTSG2nI[4]: An interrupt on compare match between the 16-bit counter value and the TSG2nCMP4 buffer register value.
- INTTSG2nI[5]: An interrupt on compare match between the 16-bit counter value and the TSG2nCMP5 buffer register value.
- INTTSG2nI[6]: An interrupt on compare match between the 16-bit counter value and the TSG2nCMP6 buffer register value.
- INTTSG2nI[7]: An interrupt on compare match between the 16-bit counter value and the TSG2nCMP7 buffer register value.
- INTTSG2nI[8]: An interrupt on compare match between the 16-bit counter value and the TSG2nCMP8 buffer register value.
- INTTSG2nI[9]: An interrupt on compare match between the 16-bit counter value and the TSG2nCMP9 buffer register value.
- INTTSG2nI[10]: An interrupt on compare match between the 16-bit counter value and the TSG2nCMP10 buffer register value.

- INTTSG2nI[11]: An interrupt on compare match between the 16-bit counter value and the TSG2nCMP11 buffer register value.
- INTTSG2nI[12]: An interrupt on compare match between the 16-bit counter value and the TSG2nCMP12 buffer register value.
- INTTSG2nIPEK: A peak interrupt when the 16-bit counter switches from incrementing to decrementing.
- INTTSG2nIVLY: A trough interrupt when the 16-bit counter switches from decrementing to incrementing (only in HT-PWM mode)
- INTTSG2nIER: An interrupt on detection of simultaneous activation of the positive phase and inverse phase
- INTTSG2nIWN: A warning detection interrupt

23.5.2 Functions of Compare Registers

The functions of the compare registers in each operating mode are shown in **Table 23.25** to **Table 23.28**.

Table 23.25 Compare Register Functions in Each Mode (1/4)

Operating Mode	TSG2nCMP0	TSG2nCMP1W	TSG2nCMP3W	TSG2nCMP5W
PWM mode	Setting PWM period	TSG2nCMP1: Setting TSOn1 clear timing TSG2nCMP2: Setting TSOn1 set timing	TSG2nCMP3: Setting TSOn2 clear timing TSG2nCMP4: Setting TSOn2 set timing	TSG2nCMP5: Setting TSOn3 clear timing TSG2nCMP6: Setting TSOn3 set timing
HT-PWM mode	Setting PWM period	TSG2nCMP1: Setting TSOn1 clear timing and TSOn2 set timing TSG2nCMP2: Setting TSOn1 set timing and TSOn2 clear timing	—	TSG2nCMP5: Setting TSOn3 clear timing and TSOn4 set timing TSG2nCMP6: Setting TSOn3 set timing and TSOn4 clear timing
SP-PWM mode	Setting PWM period	TSG2nCMP1: Setting TSOn1 clear timing and TSOn2 set timing TSG2nCMP2: Setting TSOn1 set timing and TSOn2 clear timing	—	TSG2nCMP5: Setting TSOn3 clear timing and TSOn4 set timing TSG2nCMP6: Setting TSOn3 set timing and TSOn4 clear timing
120-DC mode	Setting PWM period	TSG2nCMP1,TSG2nCMP2: Selecting TSOn1, TSOn3, and TSOn5 outputs by TSG2nPAT0.	TSG2nCMP3, TSG2nCMP4: Selecting TSOn2, TSOn4, and TSOn6 outputs by TSG2nPAT1.	TSG2nCMP5,TSG2nCMP6: Selecting TSOn1, TSOn3, and TSOn5 outputs by TSG2nPAT0.

Table 23.26 Compare Register Functions in Each Mode (2/4)

Operating Mode	TSG2nCMP7W	TSG2nCMP9W	TSG2nCMP11W	TSG2nCMP1-TSG2nCMP12
PWM mode	TSG2nCMP7: Setting TSOn4 clear timing TSG2nCMP8: Setting TSOn4 set timing	TSG2nCMP9: Setting TSOn5 clear timing TSG2nCMP10: Setting TSOn5 set timing	TSG2nCMP11: Setting TSOn6 clear timing TSG2nCMP12: Setting TSOn3 set timing	See TSG2nCMP1W to TSG2nCMP11W
HT-PWM mode	—	TSG2nCMP9: Setting TSOn5 clear timing and TSOn6 set timing TSG2nCMP10: Setting TSOn5 set timing and TSOn6 clear timing	—	See TSG2nCMP1W, TSG2nCMP5W, and TSG2nCMP9W.
SP-PWM mode	—	TSG2nCMP9: Setting TSOn5 clear timing and TSOn6 set timing TSG2nCMP10: Setting TSOn5 set timing and TSOn6 clear timing	—	See TSG2nCMP1W, TSG2nCMP5W, and TSG2nCMP9W
120-DC mode	TSG2nCMP7, TSG2nCMP8: Selecting TSOn2, TSOn4, and TSOn6 outputs by TSG2nPAT1.	TSG2nCMP9, TSG2nCMP10: Selecting TSOn1, TSOn3, and TSOn5 outputs by TSG2nPAT0	TSG2nCMP11, TSG2nCMP12: Selecting TSOn2, TSOn4, and TSOn6 outputs by TSG2nPAT1.	See TSG2nCMP1W to TSG2nCMP11W.

Table 23.27 Compare Register Functions in Each Mode (3/4)

Operating Mode	TSG2nDCMP0W	TSG2nDCMP2W	TSG2nCMPU	TSG2nCMPV
PWM mode	Setting diagnostic output or A/D conversion trigger timing	Setting diagnostic output or A/D conversion trigger timing	—	—
HT-PWM mode	Setting diagnostic output or A/D conversion trigger timing	Setting diagnostic output or A/D conversion trigger timing	The value set in TSG2nCMPU will be the TSG2nCMP1W (TSG2nCMP1, TSG2nCMP2) value.	The value set in TSG2nCMPV will be the TSG2nCMP5W (TSG2nCMP5, TSG2nCMP6) value.
SP-PWM mode	Setting diagnostic output or A/D conversion trigger timing	Setting diagnostic output or A/D conversion trigger timing	—	—
120-DC mode	Setting diagnostic output or A/D conversion trigger timing	Setting diagnostic output or A/D conversion trigger timing	—	—

Table 23.28 Compare Register Functions in Each Mode (4/4)

Operating Mode	TSG2nCMPW	TSG2nUPW	TSG2nVPW	TSG2nWPW
PWM mode	—	—	—	—
HT-PWM mode	The value set in TSG2nCMPW will be the TSG2nCMP9W (TSG2nCMP9, TSG2nCMP10) value.	—	—	—
SP-PWM mode	—	The value set in TSG2nUPW plus the value set in TSG2nCMP2 will be the TSG2nCMP1 value.	The value set in TSG2nVPW plus the value set in TSG2nCMP6 will be the TSG2nCMP5 value.	The value set in TSG2nWPW plus the value set in TSG2nCMP10 will be the TSG2nCMP9 value.
120-DC mode	—	—	—	—

23.5.3 Compare Register Rewrite Operation

The following compare registers are rewritten by reloading (TSG2nCTL3.TSG2nRMC = 0) mode or anytime rewriting (TSG2nCTL3.TSG2nRMC = 1).

The target registers are as follows:

- TSG2nCMP0
- TSG2nCMP1-TSG2nCMP12 (TSG2nCMP1W, TSG2nCMP3W, TSG2nCMP5W, TSG2nCMP7W, TSG2nCMP9W, TSG2nCMP11W)
- TSG2nPAT0W, TSG2nPAT1W
- TSG2nDTC0W, TSG2nDTC1W
- TSG2nDCMP0W, TSG2nDCMP2
- TSG2nCTL4
- TSG2nIOC3

Anytime rewriting

This mode allows direct rewriting of the values of the compare registers. That is, whenever a value is written to the compare register, the written value is reflected immediately.

CAUTION

In anytime rewriting in HT-PWM mode, if rewriting is performed again before a transfer to the buffer register is completed, the written value is not reflected immediately.

If rewriting is performed while the 16-bit counter is counting up, the value is reflected at the next peak timing of the 16-bit sub-counter. If rewriting is performed while the 16-bit counter is counting down, the value is reflected at the next trough timing of the 16-bit sub-counter.

Reloading (simultaneous rewriting)

Writing to TSG2nCMP1 (TSG2nCMP1W, TSG2nCMPU, TSG2nUPW) enables reload (sets the reload request flag (TSG2nSTR0.TSG2nRSF)), and the values of all the target registers are updated simultaneously at the next reload timing (reload).

The reload timing is the peak or trough timing of the 16-bit counter when the TSG2nTRG0.TSG2nTS bit is changed from 0 to 1. Reloading is controlled by TSG2nCTL4.TSG2nPRE and TSG2nVRE.

Writing to any register other than TSG2nCMP1 (TSG2nCMP1W, TSG2nCMPU, TSG2nUPW) does not enable reloading.

Do not write to the registers for reloading until the next reload timing after reloading is enabled by writing to TSG2nCMP1 (TSG2nCMP1W, TSG2nCMPU, TSG2nUPW). The target registers should be rewritten when the reload request flag (TSG2nSTR0.TSG2nRSF) is 0.

Rewriting the registers for reloading by DMA transfer

Some of the registers for reloading can be rewritten by DMA transfer. DMA transfer is performed as follows.

Table 23.29 Example of DMA Transfer Order of Registers for Reloading

Address	Register Name	DMA Transfer Order (Example)
<TSG2n_base1> + 040_H	TSG2nCMP1W	↑
<TSG2n_base1> + 044_H	TSG2nCMP5W	↑
<TSG2n_base1> + 048_H	TSG2nCMP9W	↑
<TSG2n_base1> + $04C_H$	TSG2nCMP3W	↑
<TSG2n_base1> + 050_H	TSG2nCMP7W	↑
<TSG2n_base1> + 054_H	TSG2nCMP11W	↑
<TSG2n_base1> + 058_H	TSG2nCMP0	↑
<TSG2n_base1> + $05C_H$	TSG2nDCMP0W	↑
<TSG2n_base1> + 060_H	TSG2nDCMP2	↑
<TSG2n_base1> + 064_H	TSG2nPAT0W	↑
<TSG2n_base1> + 068_H	TSG2nPAT1W	↑
<TSG2n_base1> + $06C_H$	TSG2nDTC0W	↑
<TSG2n_base1> + 070_H	TSG2nDTC1W	↑

Table 23.30 Duty Setting in HT-PWM Mode

Address	Register Name	DMA Transfer Order (Example)
<TSG2n_base1> + $0B0_H$	TSG2nCMPU	↑
<TSG2n_base1> + $0B4_H$	TSG2nCMPV	↑
<TSG2n_base1> + $0B8_H$	TSG2nCMPW	↑

Table 23.31 Active Width Setting in SP-PWM Mode

Address	Register Name	DMA Transfer Order (Example)
<TSG2n_base1> + $0BC_H$	TSG2nUPW	↑
<TSG2n_base1> + $0C0_H$	TSG2nVPW	↑
<TSG2n_base1> + $0C4_H$	TSG2nWPW	↑

NOTES

1. TSG2nCTL4 and TSG2nIOC3 should be rewritten individually.
2. Since writing to TSG2nCMP1W (including TSG2nCMP1, TSG2nCMPU, and TSG2nUPW) enables reloading, it should be rewritten after all the other registers for reloading have been rewritten (ready to be reloaded).

23.5.3.1 Operation Example of Anytime Rewriting

In this mode, the values written to the compare registers (TSG2nCMP1 to TSG2nCMP12) are transferred to the internal buffer registers immediately, and are compared with the counter value.

The values are transferred to the internal compare buffer registers one clock cycle (PCLK) after being written to the compare registers (TSG2nCMP1 to TSG2nCMP12).

The transfer timing of the TSG2nCMP0 is the peak or trough timing (only in HT-PWM mode) of the 16-bit counter after being written to the compare registers, or at the match timing of the TSG2nCMP0 value with the 16-bit counter value (in any mode other than HT-PWM mode).

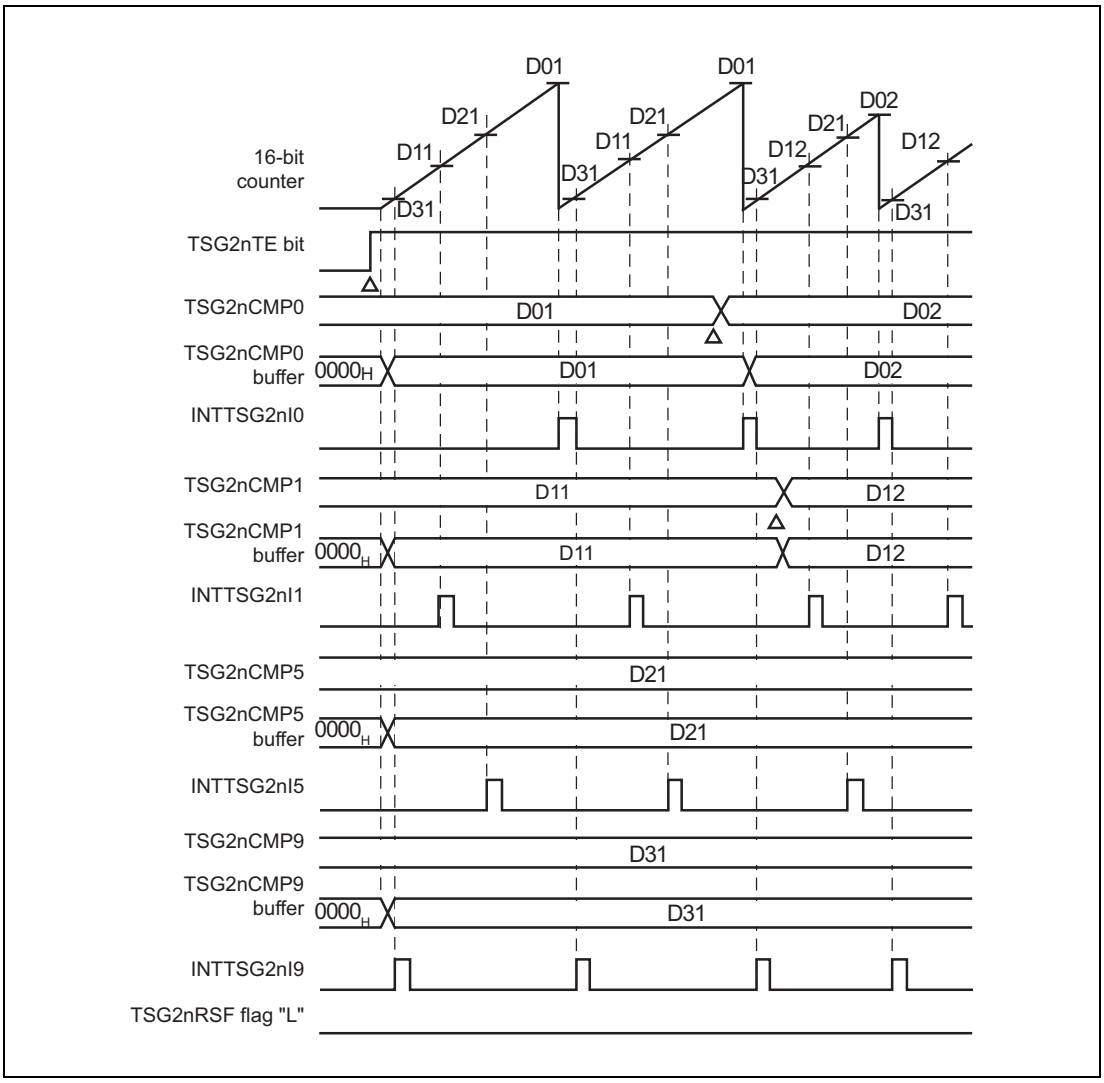


Figure 23.8 Anytime Rewriting Timing (Example in PWM Mode)

NOTES

1. D01, D02: TSG2nCMP0 setting value (0000_H to $FFFF_H$)
   D11, D12: TSG2nCMP1setting value (0000_H to $FFFF_H$)
   D21: TSG2nCMP5 setting value (0000_H to $FFFF_H$)
   D31: TSG2nCMP9 setting value (0000_H to $FFFF_H$)
2. Δ: Write access

(1) Data Reflection on PWM by Anytime Rewriting in HT-PWM Mode

In HT-PWM mode, the following output control is performed depending on the timing of anytime rewriting of the compare registers.

- If anytime rewriting is performed after the inverse phase is cleared while the 16-bit counter is counting up (the 16-bit sub-counter is counting up), the positive phase is set if the rewritten value is TSG2nCMPm < TSG2nSBC. The inverse phase is set if the rewritten value is TSG2nCMPm > TSG2nCNT. (The dead time is inserted after clearance of the positive phase and inverse phase.)
- If anytime rewriting is performed after the positive phase is set while the 16-bit counter is counting up (the 16-bit sub-counter is counting up), the positive phase is cleared if the rewritten value is TSG2nCMPm > TSG2nSBC. The positive phase is cleared and the inverse phase is set if the rewritten value is TSG2nCMPm > TSG2nCNT. (The dead time is inserted after clearance of the positive phase and inverse phase.)
- If anytime rewriting is performed before the inverse phase is cleared while the 16-bit counter is counting up (the 16-bit sub-counter is counting up), the inverse phase is cleared if the rewritten value is TSG2nCMPm < TSG2nCNT. The inverse phase is cleared and the positive phase is set if the rewritten value is TSG2nCMPm < TSG2nSBC. (The dead time is inserted after clearance of the positive phase and inverse phase.)
- If anytime rewriting is performed after the positive phase is cleared while the 16-bit counter is counting up (the 16-bit sub-counter is counting down), the inverse phase is set if the rewritten value is TSG2nCMPm > TSG2nCNT. (The dead time is inserted after clearance of the positive phase.)
- If anytime rewriting is performed before the positive phase is cleared while the 16-bit counter is counting up (the 16-bit sub-counter is counting down), the positive phase is cleared if the rewritten value is TSG2nCMPm > TSG2nSBC. The positive phase is cleared and the inverse phase is set if the rewritten value is TSG2nCMPm > TSG2nCNT. (The dead time is inserted after clearance of the positive phase and inverse phase.)
- If anytime rewriting is performed after the positive phase is set while the 16-bit counter is counting up (the 16-bit sub-counter is counting up), the positive phase is cleared if the rewritten value is TSG2nCMPm > TSG2nSBC. The positive phase is cleared and the inverse phase is set if the rewritten value is TSG2nCMPm > TSG2nCNT. (The dead time is inserted after clearance of the positive phase and inverse phase.)
- If anytime rewriting is performed after the positive phase is set while the 16-bit counter is counting down (the 16-bit sub-counter is counting up), the positive phase is cleared if the rewritten value is TSG2nCMPm > TSG2nSBC. (The dead time is inserted after clearance of the positive phase.)
- If anytime rewriting is performed after the positive phase is set while the 16-bit counter is counting down (the 16-bit sub-counter is counting up), the positive phase is cleared if the rewritten value is TSG2nCMPm > TSG2nCNT. The positive phase is cleared and the inverse phase is set if the rewritten value is TSG2nCMPm > TSG2nSBC. (The dead time is inserted after clearance of the positive phase.)

23.5.3.2 Operation Example of Reloading (Simultaneous Rewriting)

The rewritten values of the registers for reloading (TSG2nCMP0 to TSG2nCMP12, TSG2nCTL4, TSG2nIOC3, TSG2nPAT0W, TSG2nPAT1W, TSG2nDTC0W, TSG2nDTC1W, TSG2nDCMP0W, and TSG2nDCMP2) can be transferred to the corresponding buffer registers simultaneously at the reload timing.

The registers should be rewritten when the pertinent reload request flag (TSG2nSTR0.TSG2nRSF) is 0.

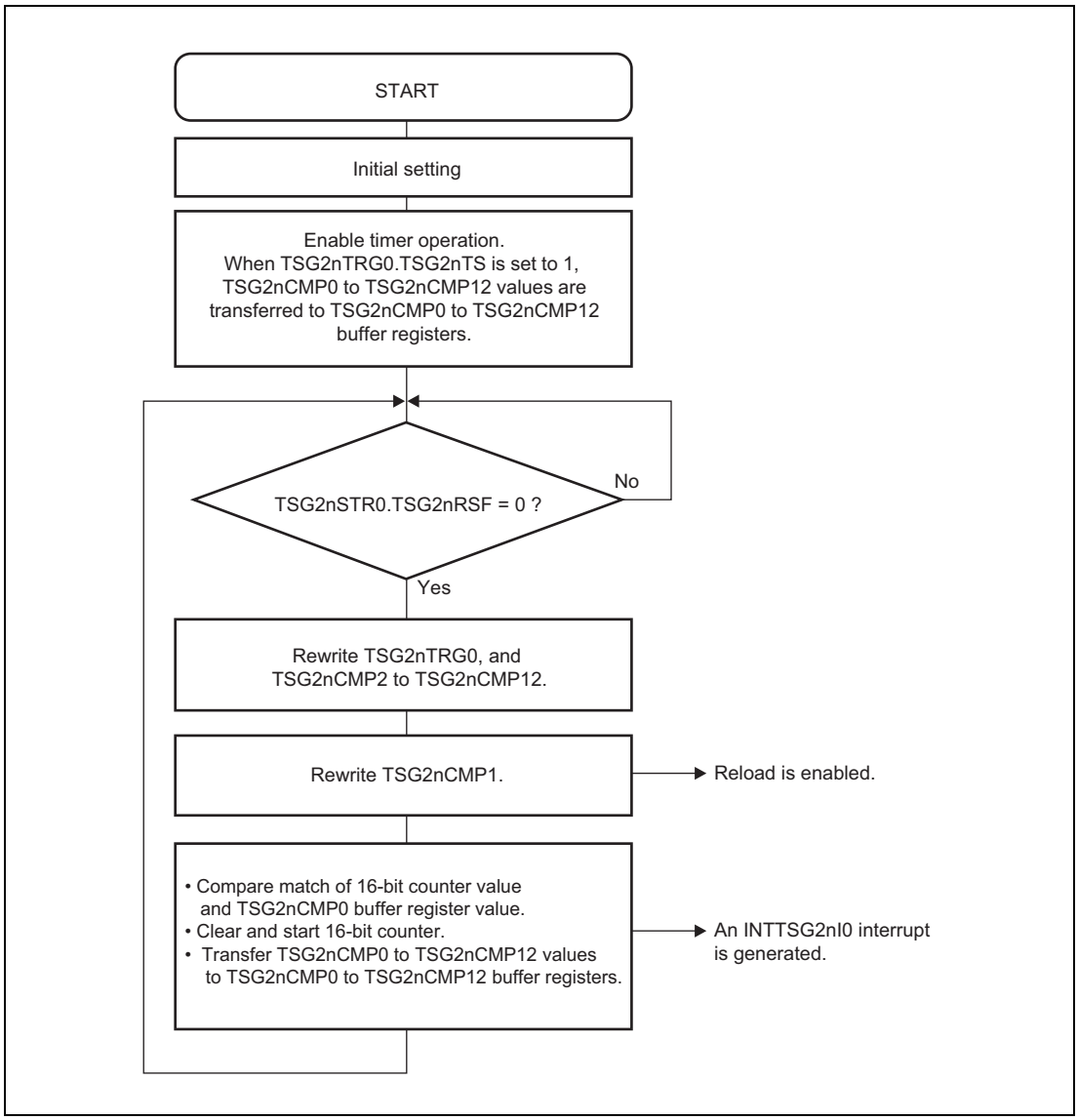


Figure 23.9 Basic Operation Flow of Reloading (Simultaneous Rewriting) (Example of PWM Mode)

CAUTION

Writing to TSG2nCMP1 also enables reloading. Therefore, TSG2nCMP1 should be rewritten after TSG2nCMP0 and TSG2nCMP2 to TSG2nCMP12 registers have been rewritten.

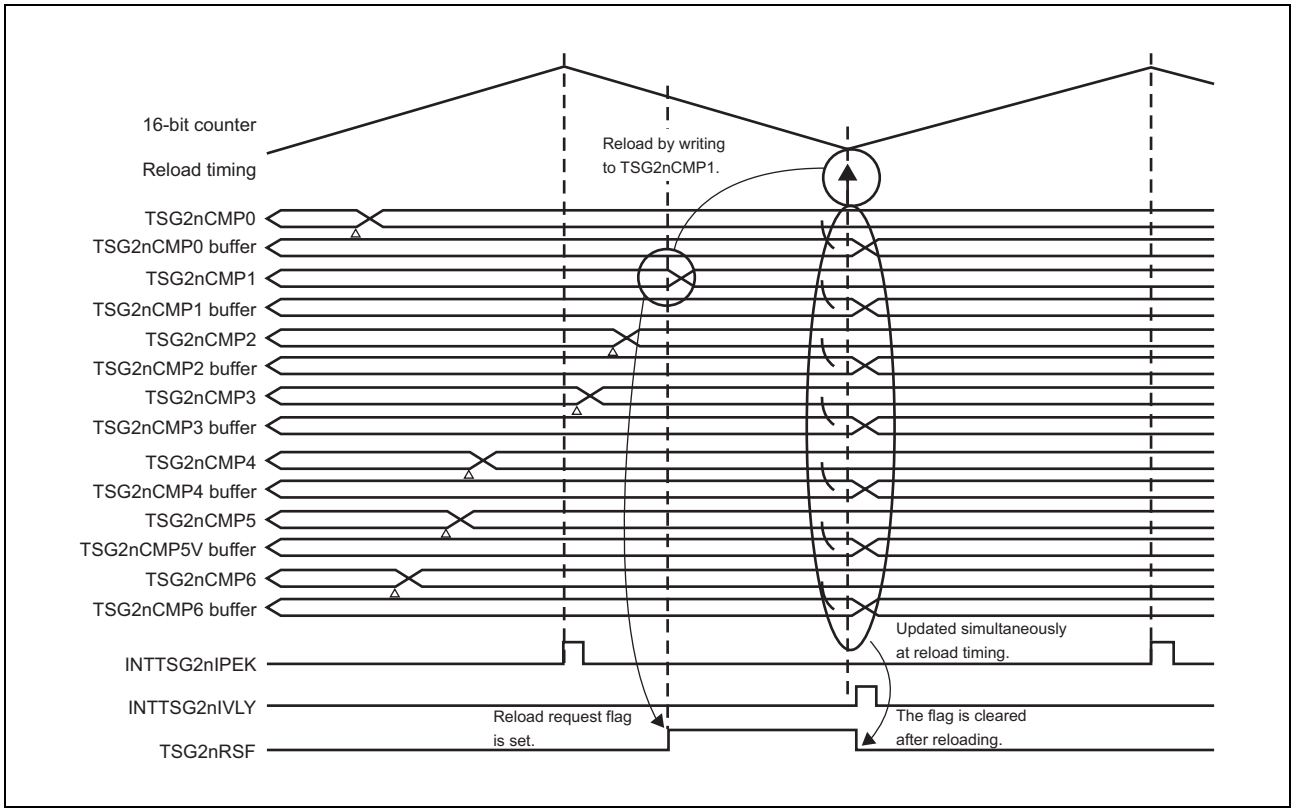


Figure 23.10 Simultaneous Rewriting Timing (1/2) (Example of HT-PWM Mode)

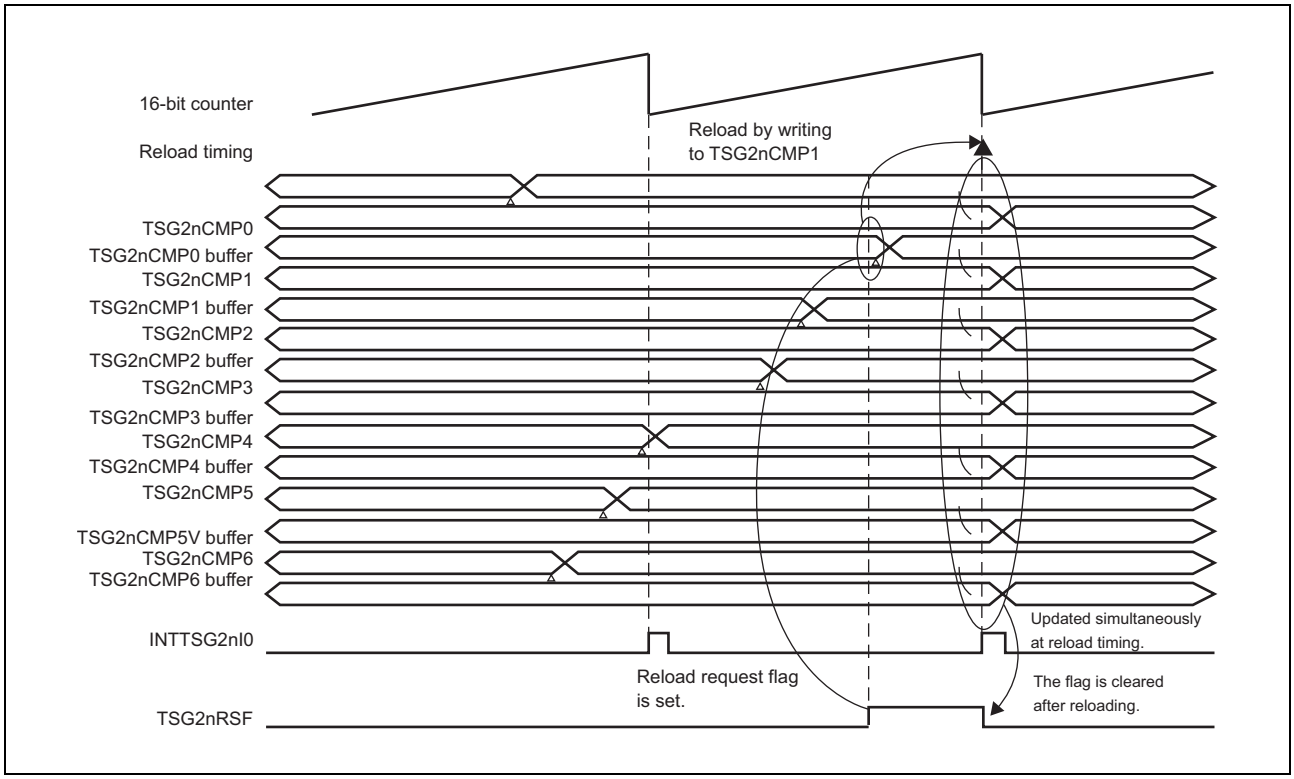


Figure 23.11 Simultaneous Rewriting Timing (2/2) (Example of PWM mode)

(1) Reload Setting Example in Each Mode

Reloading conditions and setting examples are shown below.

Table 23.32 List of Reload Settings (when TSG2nCTL3.TSG2nRIA = 0)

Operating Mode	TSG2nCTL4.TSG2nPRE	TSG2nCTL4.TSG2nVRE	TSG2nCTL4.TSG2nPIE	TSG2nCTL4.TSG2nVIE	TSG2nCTL4.TSG2nRCC04-TSG2nRCC00	Reload
PWM mode SP-PWM mode 120-DC mode	0	0/1	0/1	0/1	Any value	Setting is prohibited
	1	0	0/1	0/1	Any value	When INTTSG2nI[0] is generated.
	1	1	0/1	0/1	Any value	When INTTSG2nI[0] is generated.
HT-PWM mode	0	0	0/1	0/1	Any value	Setting is prohibited
	0	1	0/1	0/1	Any value	When INTTSG2nIVLY is generated.
	1	0	0/1	0/1	Any value	When INTTSG2nIPEK is generated.
	1	1	0/1	0/1	Any value	When INTTSG2nIPEK or INTTSG2nIVLY is generated.

Table 23.33 List of Reload Settings (when TSG2nCTL3.TSG2nRIA = 1)

Operating Mode	TSG2nCTL4.TSG2nPRE	TSG2nCTL4.TSG2nVRE	TSG2nCTL4.TSG2nPIE	TSG2nCTL4.TSG2nVIE	TSG2nCTL4.TSG2nRCC04-TSG2nRCC00	Reload
PWM mode SP-PWM mode 120-DC mode	0	0/1	0/1	0/1	Any value	Setting is prohibited
	1	0	0	0/1	Any value	Setting is prohibited
	1	0	1	0/1	Any value	When INTTSG2nI[0] is generated.
	1	1	0	0/1	Any value	Setting is prohibited
	1	1	1	0/1	Any value	When INTTSG2nI[0] is generated.
HT-PWM mode	0	0	0/1	0/1	Any value	Setting is prohibited
	0	1	0	0	Any value	Setting is prohibited
	0	1	0	1	Any value	When INTTSG2nIVLY is generated.
	0	1	1	0	Any value	Setting is prohibited
	0	1	1	1	Any value	When INTTSG2nIVLY is generated.
	1	0	0	0/1	Any value	Setting is prohibited
	1	0	1	0/1	Any value	When INTTSG2nIPEK is generated.
	1	1	0	0	Any value	Setting is prohibited
	1	1	0	1	Any value	When INTTSG2nIVLY is generated.
	1	1	1	0	Any value	When INTTSG2nIPEK is generated.
	1	1	1	1	Any value	When INTTSG2nIPEK/ INTTSG2nIVLY is generated.

23.5.4 List of Outputs in Each Mode

23.5.4.1 Timer Output in Each Mode

The list of timer outputs (TSOn0 to TSOn7) in each mode is shown in **Table 23.34** to **Table 23.36**.

Table 23.34 List of Timer Outputs in Each Mode (1/3)

Operating Mode	TSOn0 Pin	TSOn1 Pin	TSOn2 Pin
PWM mode	— (fixed to low.)	Outputs a PWM signal by compare match of TSG2nCMP1W (TSG2nCMP1 and TSG2nCMP2).	Outputs a PWM signal by compare match of TSG2nCMP3W (TSG2nCMP3 and TSG2nCMP4).
HT-PWM mode	Outputs the status indicating whether the 16-bit counter or 16-bit sub-counter is incremented or decremented.	Outputs a positive phase PWM signal (with dead time) by compare match of TSG2nCMP1W (TSG2nCMP1 and TSG2nCMP2).	Outputs an inverse phase PWM signal (with dead time) to the TSOn1 pin.
SP-PWM mode	— (fixed to low.)	Outputs a positive phase PWM signal (with dead time) by compare match of TSG2nCMP1W (TSG2nCMP1 and TSG2nCMP2).	Outputs an inverse phase PWM signal (with dead time) to the TSOn1 pin.
120-DC mode	— (fixed to low.)	Outputs a PWM signal by TSG2nCMP1W (TSG2nCMP1 and TSG2nCMP2), TSG2nCMP5W (TSG2nCMP5 and TSG2nCMP6), and TSG2nCMP9W (TSG2nCMP9 and TSG2nCMP10).	Outputs a PWM signal by TSG2nCMP3W (TSG2nCMP3 and TSG2nCMP4), TSG2nCMP7W (TSG2nCMP7 and TSG2nCMP8), and TSG2nCMP11W (TSG2nCMP11 and TSG2nCMP12).

Table 23.35 List of Timer Outputs in Each Mode (2/3)

Operating Mode	TSOn3 Pin	TSOn4 Pin	TSOn5 Pin
PWM mode	Outputs a PWM signal by compare match of TSG2nCMP5W (TSG2nCMP5 and TSG2nCMP6).	Outputs a PWM signal by compare match of TSG2nCMP7W (TSG2nCMP7 and TSG2nCMP8).	Outputs a PWM signal by compare match of TSG2nCMP9W (TSG2nCMP9 and TSG2nCMP10).
HT-PWM mode	Outputs a positive phase PWM signal (with dead time) by compare match of TSG2nCMP5W (TSG2nCMP5 and TSG2nCMP6).	Outputs an inverse phase PWM signal (with dead time) to the TSOn3 pin.	Outputs a positive phase PWM signal (with dead time) by compare match of TSG2nCMP9W (TSG2nCMP9 and TSG2nCMP10).
SP-PWM mode	Outputs a positive phase PWM signal (with dead time) by compare match of TSG2nCMP5W (TSG2nCMP5 and TSG2nCMP6).	Outputs an inverse phase PWM signal (with dead time) to the TSOn3 pin.	Outputs a positive phase PWM signal (with dead time) by compare match of TSG2nCMP9W (TSG2nCMP9 and TSG2nCMP10).
120-DC mode	Outputs a PWM signal by TSG2nCMP1W (TSG2nCMP1 and TSG2nCMP2), TSG2nCMP5W (TSG2nCMP5 and TSG2nCMP6), and TSG2nCMP9W (TSG2nCMP9 and TSG2nCMP10).	Outputs a PWM signal by TSG2nCMP3W (TSG2nCMP3 and TSG2nCMP4), TSG2nCMP7W (TSG2nCMP7 and TSG2nCMP8), and TSG2nCMP11W (TSG2nCMP11 and TSG2nCMP12).	Outputs a PWM signal by TSG2nCMP1W (TSG2nCMP1 and TSG2nCMP2), TSG2nCMP5W (TSG2nCMP5 and TSG2nCMP6), and TSG2nCMP9W (TSG2nCMP9 and TSG2nCMP10).

Table 23.36 List of Timer Outputs in Each Mode (3/3)

Operating Mode	TSOn6 Pin	TSOn7 Pin
PWM mode	Outputs a PWM signal by compare match of TSG2nCMP11W (TSG2nCMP11 and TSG2nCMP12).	Outputs a diagnostic signal or A/D conversion trigger.
HT-PWM mode	Outputs an inverse phase PWM signal (with dead time) to the TSOn5 pin.	Outputs a diagnostic signal or A/D conversion trigger.
SP-PWM mode	Outputs an inverse phase PWM signal (with dead time) to the TSOn5 pin.	Outputs a diagnostic signal or A/D conversion trigger*1.
120-DC mode	Outputs a PWM signal by TSG2nCMP3W (TSG2nCMP3 and TSG2nCMP4), TSG2nCMP7W (TSG2nCMP7 and TSG2nCMP8), and TSG2nCMP11W (TSG2nCMP11 and TSG2nCMP12).	Outputs a diagnostic signal or A/D conversion trigger*1.

Note 1. For TSOn7, see **Section 23.5.4.1**, **(1) TSOn7 Pin Output Control**.

(1) TSOn7 Pin Output Control

The TSOn7 pin can output an A/D conversion trigger pulse (TSG2nIOC1.TSG2nTGS = 0) or diagnostic signal (TSG2nIOC1.TSG2nTGS = 1). When outputting an A/D conversion trigger pulse, the TSOn7 pin is activated at the rising edge of the TSTADT0 signal, and inactivated at the rising edge of the TSTADT1 signal. When the TSTADT0 signal is detected while the TSOn7 pin is active, the TSOn7 pin remains active. When the TSTADT1 signal is detected while the TSOn7 pin is inactive, the TSOn7 pin remains inactive. If TSTADT0 and TSTADT1 signal triggers occur simultaneously, the TSOn7 pin is inactivated.

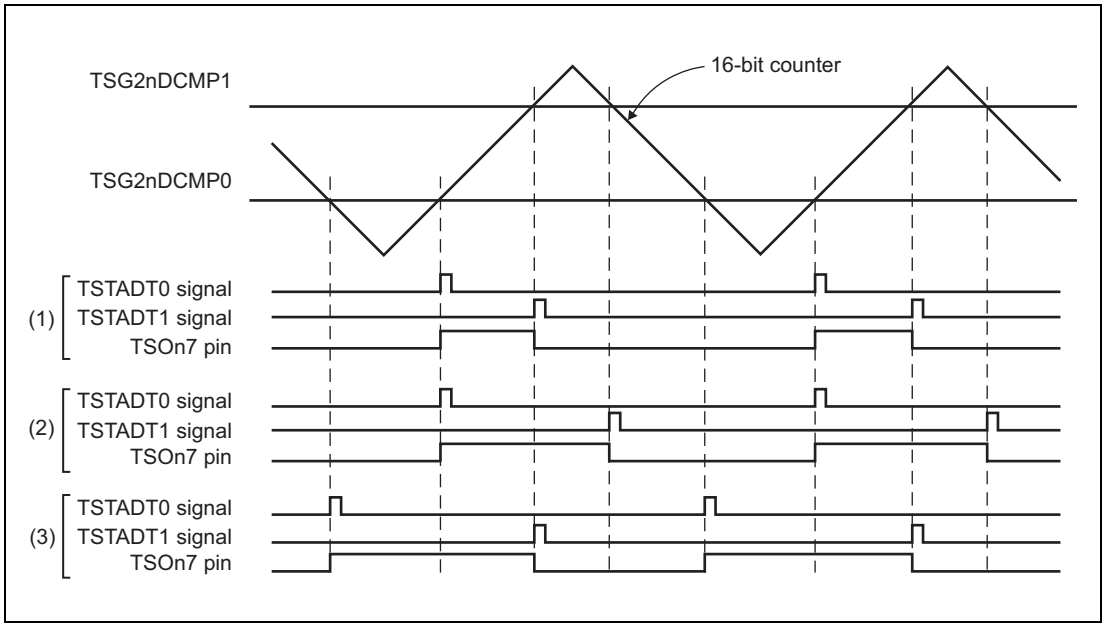


Figure 23.12 Timing Example of A/D Trigger Output by the TSOn7 Pin (TSG2nIOC1.TSG2nTGS = 0)

NOTES

1. When TSG2nDCMP0 < TSG2nDCMP1, TSG2nCTL5 = 0004_H, and TSG2nCTL6 = 0010_H
2. When TSG2nDCMP0 < TSG2nDCMP1, TSG2nCTL5 = 0004_H, and TSG2nCTL6 = 0020_H
3. When TSG2nDCMP0 < TSG2nDCMP1, TSG2nCTL5 = 0008_H, and TSG2nCTL6 = 0010_H
   For TSOn7, see **Section 23.5.4.1 (1) TSOn7 Pin Output Control**.

The TSOn7 pin during diagnostic output outputs the active level with the output width specified by TSG2nCTL0.TSG2nDWD at the match timing of the TSG2nDCMP0 to TSG2nDCMP2 values with the 16-bit counter value. If another match of the TSG2nDCMP0 to TSG2nDCMP2 values with the 16-bit counter value occurs consecutively within the output width specified by TSG2nDWD causing their active level widths to be overlapped, the TSOn7 pin outputs a pulse within 16 clock cycles (PCLK).

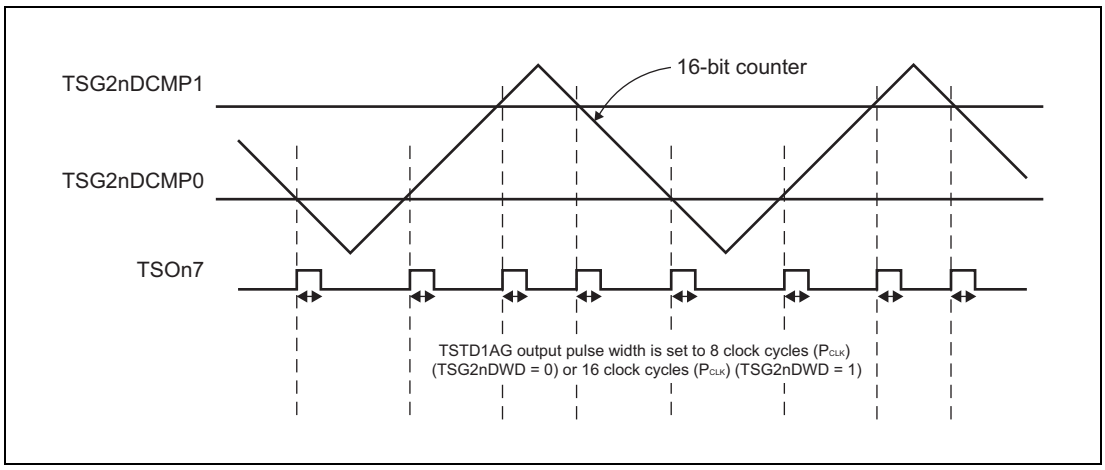


Figure 23.13 Timing Example of Diagnostic Output by the TSOn7 Pin (1) (TSG2nIOC1.TSG2nTGS = 1)

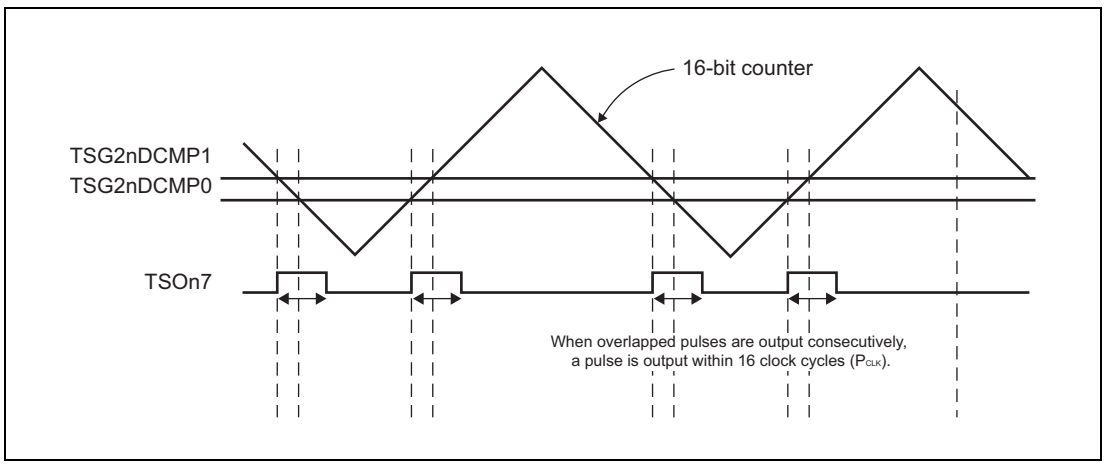


Figure 23.14 Timing Example of Diagnostic Output by the TSOn7 Pin (2) (with Output Width Overlapped)

23.5.4.2 Interrupts in Each Mode

A list of interrupts (INTTSG2nI[0] to INTTSG2nI[12], INTTSG2nIPEK, INTTSG2nIVLY, INTTSG2nIER and INTTSG2nIWN) in each mode is shown in **Table 23.37** to **Table 23.41**.

Table 23.37 List of Interrupts in Each Mode (1/5)

Operating mode	INTTSG2nI[0]	INTTSG2nI[1]	INTTSG2nI[2]	INTTSG2nI[3]
PWM mode	TSG2nCMP0 compare match interrupt	TSG2nCMP1 compare match interrupt*1	TSG2nCMP2 compare match interrupt*1	TSG2nCMP3 compare match interrupt*1
HT-PWM mode	Periodic interrupt	TSG2nCMP1 compare match interrupt*2	TSG2nCMP2 compare match interrupt*2	—
SP-PWM mode	TSG2nCMP0 compare match interrupt	TSG2nCMP1 compare match interrupt*1	TSG2nCMP2 compare match interrupt *1	—
120-DC mode	TSG2nCMP0 compare match interrupt	TSG2nCMP1 compare match interrupt*1	TSG2nCMP2 compare match interrupt*1	TSG2nCMP3 compare match interrupt *1

Note 1. When TSG2nCMP0 < TSG2nCMPm, a compare match interrupt is not generated (m = 1 to 12).
Note 2. When $0000_H \leq$ TSG2nCMPm < TSG2nDTC0, and (TSG2nCMP0 + TSG2nDTC0) < TSG2nCMPm, a compare match interrupt is not generated.

Table 23.38 List of Interrupts in Each Mode (2/5)

Operating mode	INTTSG2nI[4]	INTTSG2nI[5]	INTTSG2nI[6]	INTTSG2nI[7]
PWM mode	TSG2nCMP4 compare match interrupt*1	TSG2nCMP5 compare match interrupt*1	TSG2nCMP6 compare match interrupt*1	TSG2nCMP7 compare match interrupt*1
HT-PWM mode	—	TSG2nCMP5 compare match interrupt*2	TSG2nCMP6 compare match interrupt*2	—
SP-PWM mode	—	TSG2nCMP5 compare match interrupt*1	TSG2nCMP6 compare match interrupt*1	—
120-DC mode	TSG2nCMP4 compare match interrupt*1	TSG2nCMP5 compare match interrupt*1	TSG2nCMP6 compare match interrupt*1	TSG2nCMP7 compare match interrupt*1

Note 1. When TSG2nCMP0 < TSG2nCMPm, a compare match interrupt is not generated (m = 1 to 12).
Note 2. When $0000_H \leq$ TSG2nCMPm < TSG2nDTC0, and (TSG2nCMP0 + TSG2nDTC0) < TSG2nCMPm, a compare match interrupt is not generated.

Table 23.39 List of Interrupts in Each Mode (3/5)

Operating mode	INTTSG2nI[8]	INTTSG2nI[9]	INTTSG2nI[10]	INTTSG2nI[11]
PWM mode	TSG2nCMP8 compare match interrupt*1	TSG2nCMP9 compare match interrupt*1	TSG2nCMP10 compare match interrupt*1	TSG2nCMP11 compare match interrupt*1
HT-PWM mode	—	TSG2nCMP9 compare match interrupt*2	TSG2nCMP10 compare match interrupt*2	—
SP-PWM mode	—	TSG2nCMP9 compare match interrupt*1	TSG2nCMP10 compare match interrupt*1	—
120-DC mode	TSG2nCMP8 compare match interrupt*1	TSG2nCMP9 compare match interrupt*1	TSG2nCMP10 compare match interrupt*1	TSG2nCMP11 compare match interrupt*1

Note 1. When TSG2nCMP0 < TSG2nCMPm, a compare match interrupt is not generated (m = 1 to 12).
Note 2. When $0000_H \leq$ TSG2nCMPm < TSG2nDTC0, and (TSG2nCMP0 + TSG2nDTC0) < TSG2nCMPm, a compare match interrupt is not generated.

Table 23.40 List of Interrupts in Each Mode (4/5)

Operating mode	INTTSG2nI[12]	INTTSG2nIPEK	INTTSG2nIVLY
PWM mode	TSG2nCMP12 compare match interrupt*1	Peak interrupt at the same timing as INTTSG2nI[0]	—
HT-PWM mode	—	Peak interrupt	Trough interrupt
SP-PWM mode	—	Peak interrupt at the same timing as INTTSG2nI[0]	—
120-DC mode	TSG2nCMP12 compare match interrupt*1	Peak interrupt at the same timing as INTTSG2nI[0]	—

Note 1. When TSG2nCMP0 < TSG2nCMPm, a compare match interrupt is not generated (m = 1 to 12).

Table 23.41 List of Interrupts in Each Mode (5/5)

Operating mode	INTTSG2nIER	INTTSG2nIWN
PWM mode	Error interrupt	Warning interrupt
HT-PWM mode	Error interrupt	Warning interrupt
SP-PWM mode	Error interrupt	Warning interrupt
120-DC mode	Error interrupt	Warning interrupt

23.6 Match Interrupt

TSG2n can interrupts include a compare match interrupt (INTTSG2nIm), a peak interrupt (INTTSG2nIPEK), and a trough interrupt (INTTSG2nIVLY). For error interrupt and warning interrupts (INTTSG2nIER and INTTSG2nIWN), see **Section 23.10, Error and Warning Interrupts**.

A periodic interrupt (INTTSG2nI[0]) is generated for each timer cycle. In HTPWM mode, it is generated when the TSG2nDTC0 buffer register value matches with the 16-bit counter value. When the 16-bit counter performs sawtooth wave operation (PWM mode, SP-PWM mode, and 120-DC mode), it is generated after the 16-bit counter value has matched with the TSG2nCMP0 buffer register value.

A compare match interrupt (INTTSG2nIm) is generated on matches between the TSG2nCMPm buffer register value and the 16-bit counter value depending on the compare register to be used in each operating mode (m = 1 to 12).

A peak interrupt (INTTSG2nIPEK) is generated in all modes. In HT-PWM mode, it is generated when the 16-bit counter switches from incrementing to decrementing. When the 16-bit counter performs sawtooth wave operation (PWM mode, SP-PWM mode, and 120-DC mode), it is generated after the 16-bit counter value has matched with the TSG2nCMP0 buffer register value (the same timing as an INTTSG2nI[0] interrupt).

A trough interrupt (INTTSG2nIVLY) is generated when the 16-bit counter switches from decrementing to incrementing in HT-PWM mode.

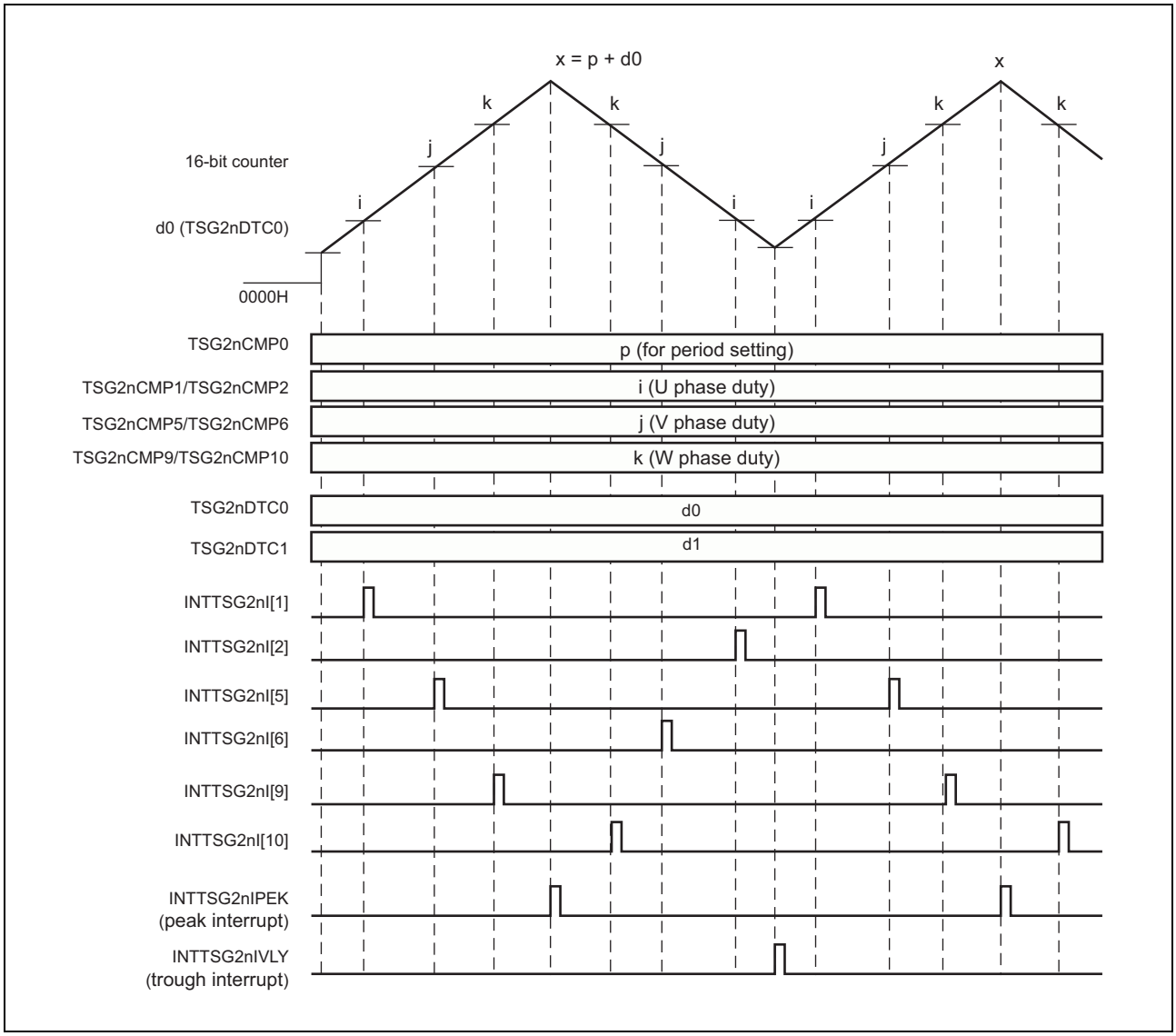


Figure 23.15 Interrupt Generation Example (1/2) (Example of HT-PWM Mode)

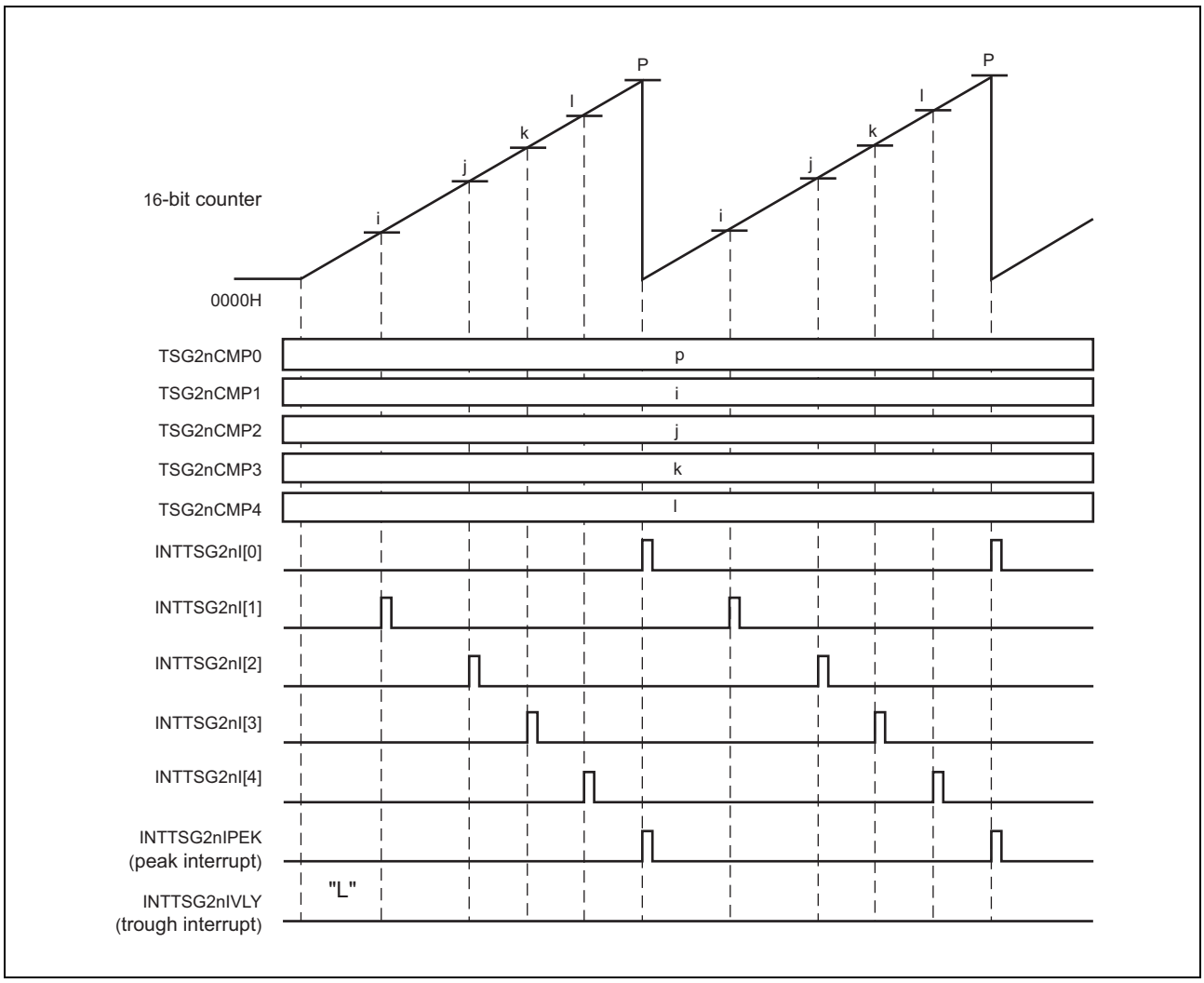


Figure 23.16 Interrupt Generation Example (2/2) (Example of PWM Mode)

23.7 Flags

Table 23.42 List of Flags

No.	Flag Name	Symbol	Register	Operating Mode
(1)	Up-count flag	TSG2nCUF	TSG2nSTR0	HT-PWM mode
		TSG2nSUF	TSG2nSTR0	
(2)	Positive/negative phase simultaneous activation detection flag	TSG2nTBF0-TSG2nTBF2	TSG2nSTR2	All operating modes
(3)	Reload request flag	TSG2nRSF	TSG2nSTR0	All operating modes
(4)	Noise detection flag	TSG2nNDF	TSG2nSTR2	All operating modes
(5)	Pattern order detection flag	TSG2nTSF	TSG2nSTR1	All operating modes
(6)	Pattern error detection flag	TSG2nPEF	TSG2nSTR2	All operating modes
(7)	Pattern inversion detection flag	TSG2nPRF	TSG2nSTR2	All operating modes
(8)	Pattern phase difference detection flag	TSG2nPPF	TSG2nSTR2	All operating modes
(9)	Timer output pattern flag	TSG2nOPF0-TSG2nOPF2	TSG2nSTR1	All operating modes
(10)	Pattern switch detection signal (internal signal)	TSG2nPTE	—	All operating modes

23.7.1 Up-Count Flags (TSG2nCUF and TSG2nSUF)

Name

Up count flag (TSG2nSTR0.TSG2nCUF and TSG2nSUF)

Description

There are following two up-count flags.

TSG2nCUF is an up-/down-count flag of the 16-bit counter.

TSG2nSUF is an up-/down-count flag of the 16-bit sub-counter.

For both TSG2nCUF and TSG2nSUF, 0 indicates incrementing, whereas 1 indicates decrementing.

TSG2nCUF and TSG2nSUF can be used only in HT-PWM mode.

Example of operation

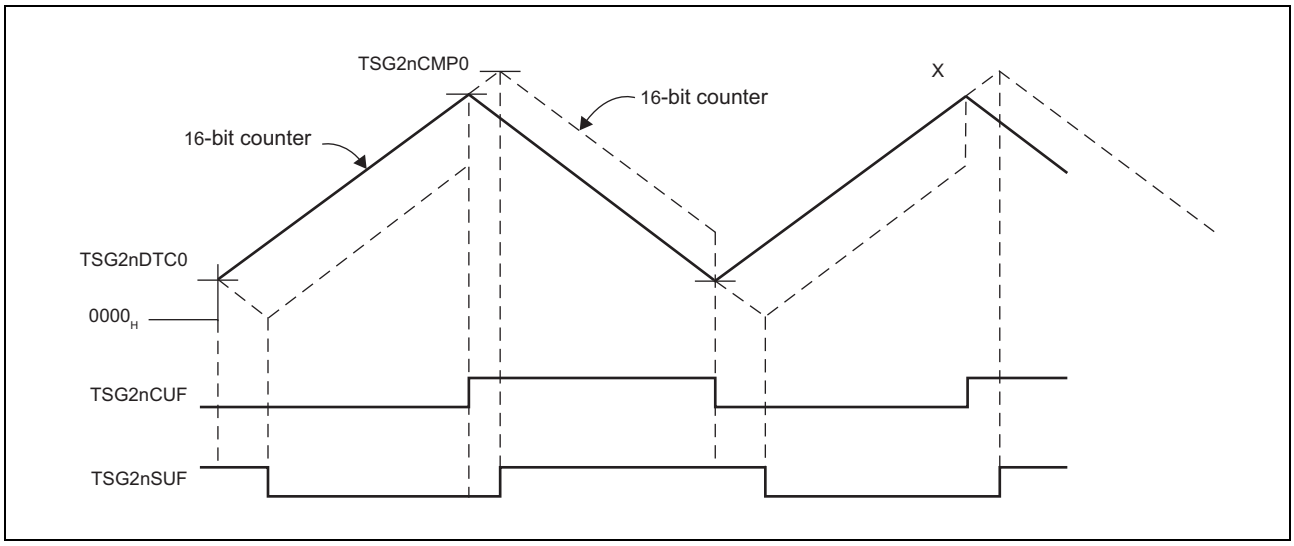


Figure 23.17 Example of Up-Count Flag Operation

NOTES

1. TSG2nCUF is set to the value as below:
   0 (up-count) when TSG2nDTC0 ≤ 16-bit counter ≤ (TSG2nCMP0 + TSG2nDTC0 - 2)
   1 (down-count) when (TSG2nCMP0 + TSG2nDTC0) ≥ 16-bit counter ≥ TSG2nDTC0 + 2
2. TSG2nSUF is set to the value as below:
   0 (up-count) when 0 ≤ 16-bit sub-counter ≤ (TSG2nCMP0 + TSG2nDTC0 + TSG2nDTC1 - 2)
   1 (down-count) when (TSG2nCMP0 + TSG2nDTC0 + TSG2nDTC1) ≥ 16-bit sub-counter ≥ 2

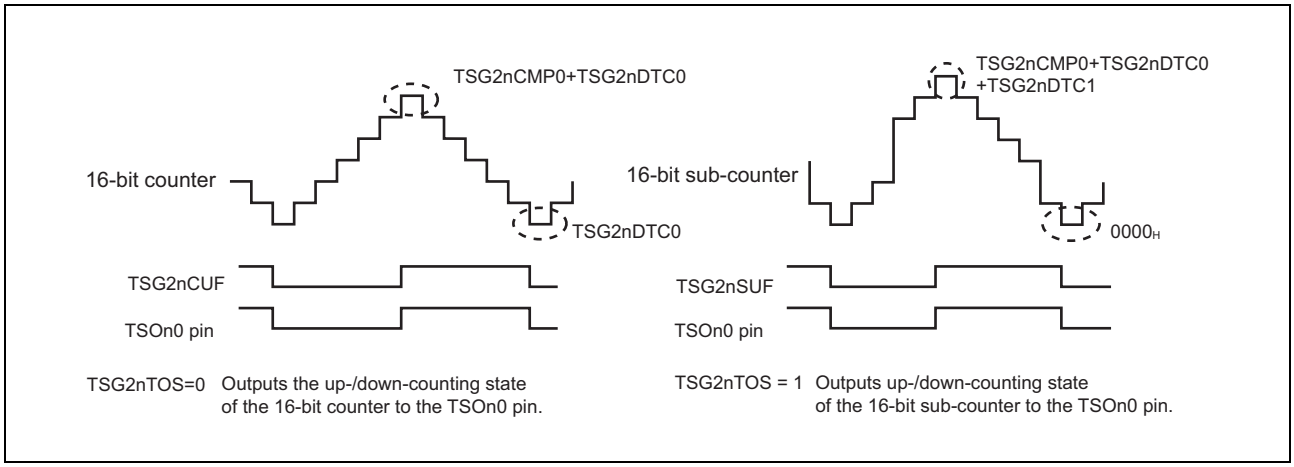


Figure 23.18 TSOn0 Pin Output depending onTSG2nIOC1.TSG2nTOS Setting

Operating mode

TSG2nCUF and TSG2nSUF can be used only in HT-PWM mode.

23.7.2 Positive/Negative Phase Simultaneous Activation Detection Flags (TSG2nTBF0 to TSG2nTBF2)

Name

Positive phase and negative phase simultaneous activation detection flags (TSG2nSTR2.TSG2nTBF0 to TSG2nTBF2 flags)

Description

When any of TSG2nCTL1.TSG2nTBA2 to TSG2nTBA0 is 1, TSG2nTBF0 to TSG2nTBF2 can detect simultaneous activation of the positive and negative phases of TSG2n.

When the simultaneous active state of the positive phase and inverse phase of the TSG2n is detected, the corresponding TSG2nTBF0 to TSG2nTBF2 flags are set to 1, and an error interrupt (INTTSG2nIER) is generated. The flags are cleared when 1 is written to TSG2nSTC.TSG2nTBR0 to TSG2nTBR2, respectively.

Example of operation

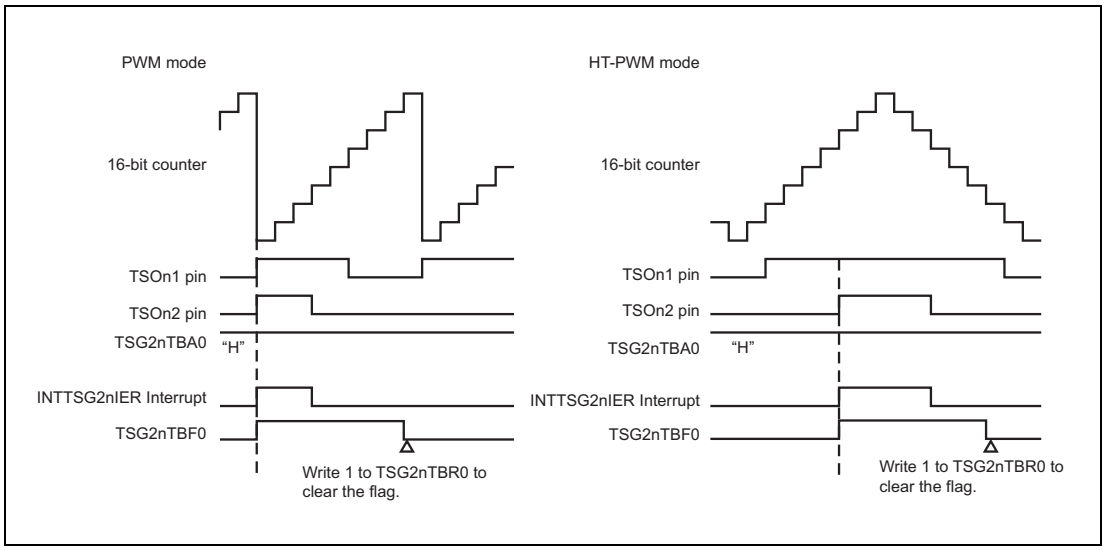


Figure 23.19 Example of Positive /Negative Phase Simultaneous Activation Detection Flag Operation

Operating mode

TSG2nTBF0 to TSG2nTBF2 can be used in all operating modes.

CAUTION

TSG2nTBF0 to TSG2nTBF2 are valid only when TSG2nCTL1.TSG2nTBA0 to TSG2nTBA2 = 1 and TSG2nSTR0.TSG2nTE = 1.

23.7.3 Reload Request Flag (TSG2nRSF)

Name

Reload request flag (TSG2nSTR0.TSG2nRSF)

Description

TSG2nRSF is set to 1 when a reload request is generated (when a value is written to TSG2nCMP1 (TSG2nCMP1W, TSG2nCMPU, TSG2nCPW)), and cleared to 0 when the value is transferred to all the buffer registers.

Example of operation

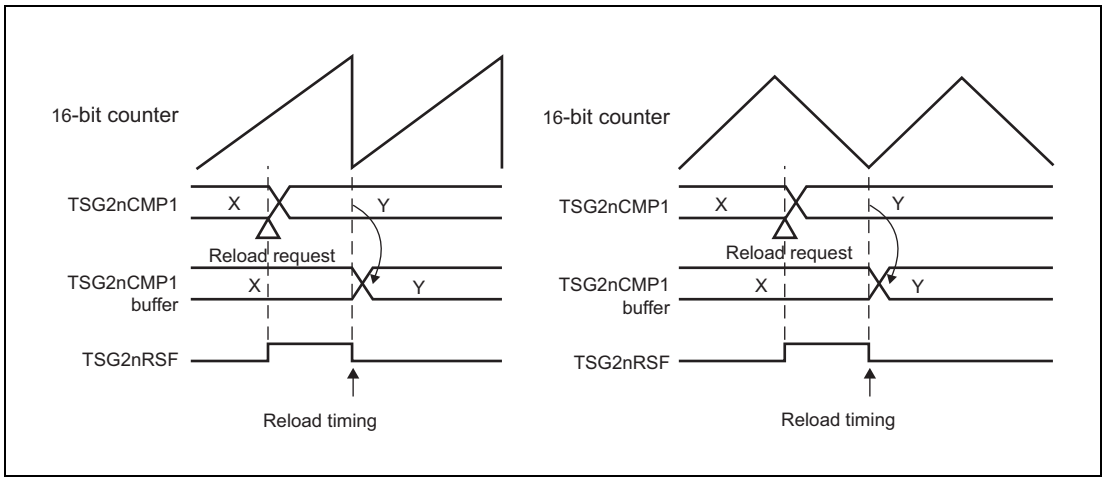


Figure 23.20 Example of Reload Request Flag Operation

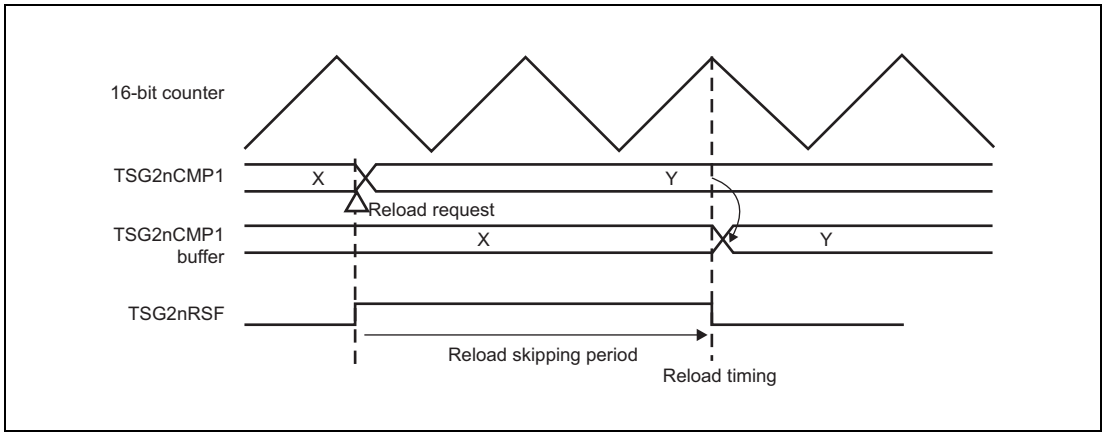


Figure 23.21 Reload Request Flag and Reload Skipping Period

Operating Mode

TSG2nRSF can be used in all operating modes.

23.7.4 Noise Detection Flag (TSG2nNDF)

Name

Noise detection flag (TSG2nSTR2.TSG2nNDF)

Description

TSG2nNDF can detect that two or more pins of TAPTSn2 to TAPTSn0 have changed simultaneously (a noise is generated).

TSG2nNDF is set to 1 when two or more pins of TAPTSn2 to TAPTSn0 have changed simultaneously (a noise is generated), and a warning interrupt (INTTSG2nIWN) is generated. The TSG2nNDF flag is cleared to 0 when 1 is written to the TSG2nSTC.TSG2nNDR bit.

Example of operation

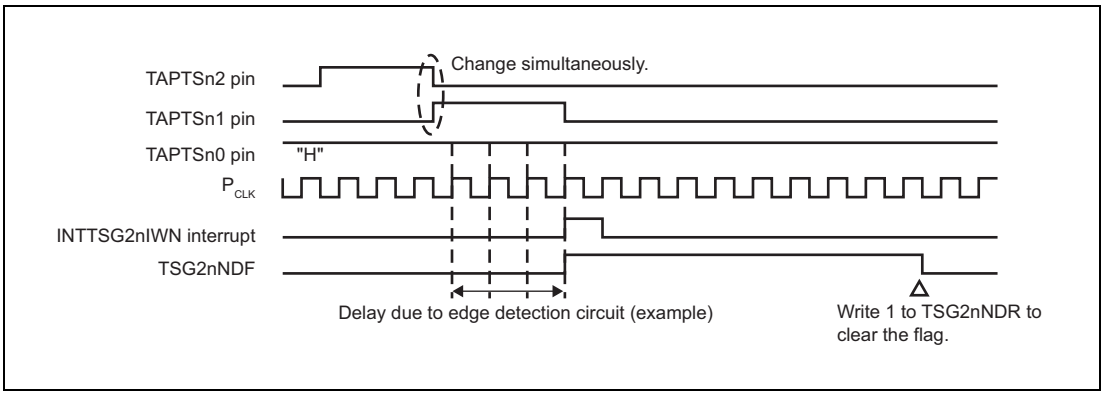


Figure 23.22 Example of Noise Detection Flag Operation

Operating mode

TSG2nNDF can be used in all operating modes.

CAUTION

TSG2nNDF is valid only when TSG2nCTL1.TSG2nNDC = 1 and TSG2nSTR0.TSG2nTE = 1.

23.7.5 Pattern Order Detection Flag (TSG2nTSF)

Name

Pattern order detection flag (TSG2nSTR1.TSG2nTSF)

Description

TSG2nTSF can detect the order of patterns input to the TAPTSn2 to TAPTSn0 pins.

TSG2nTSF is set depending on the values input to the TAPTSn2 to TAPTSn0 pins as shown in the table below.

TSG2nTSF	Values input to TAPTSn2 to TAPTSn0 Pins
0	[1,0,1] → [1,0,0] → [1,1,0] → [0,1,0] → [0,1,1] → [0,0,1]
1	[1,0,1] ← [1,0,0] ← [1,1,0] ← [0,1,0] ← [0,1,1] ← [0,0,1]

Example of operation

23.7.5.1 When Normal Input to TAPTSn2 to TAPTSn0 Pins is Detected

As shown in **Figure 23.23**, if the TAPTSn2 to TAPTSn0 pins change in the normal order, 0 or 1 is set according to the change order at the change timing.

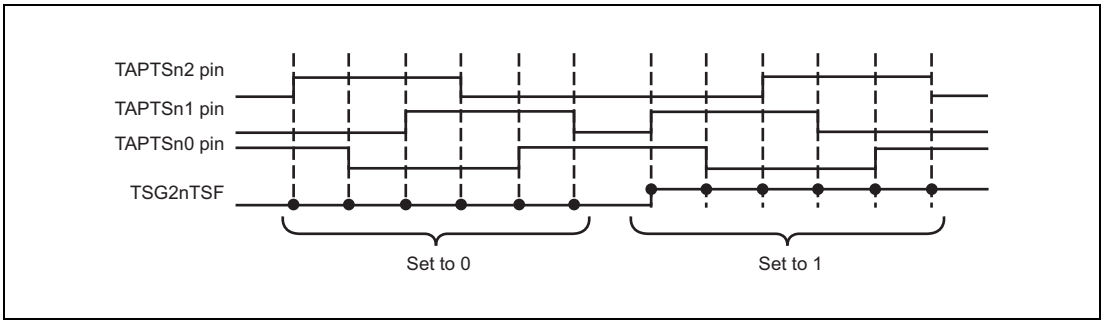


Figure 23.23 Example of Pattern Order Detection Flag Operation (Normal Operation)

23.7.5.2 Detection of Input Pattern Change

Immediately after TSG2n starts operation, the rotation direction cannot be determined. Therefore, TSG2nTSF cannot detect a change (normal or reverse rotation) in the pattern input to the TAPTSn2 to TAPTSn0 pins. To enable detection of a change immediately after the start of operation, TSG2nPSC should be set before operation starts (when TSG2nTE = 0, the TSG2nPSC value is reflected).

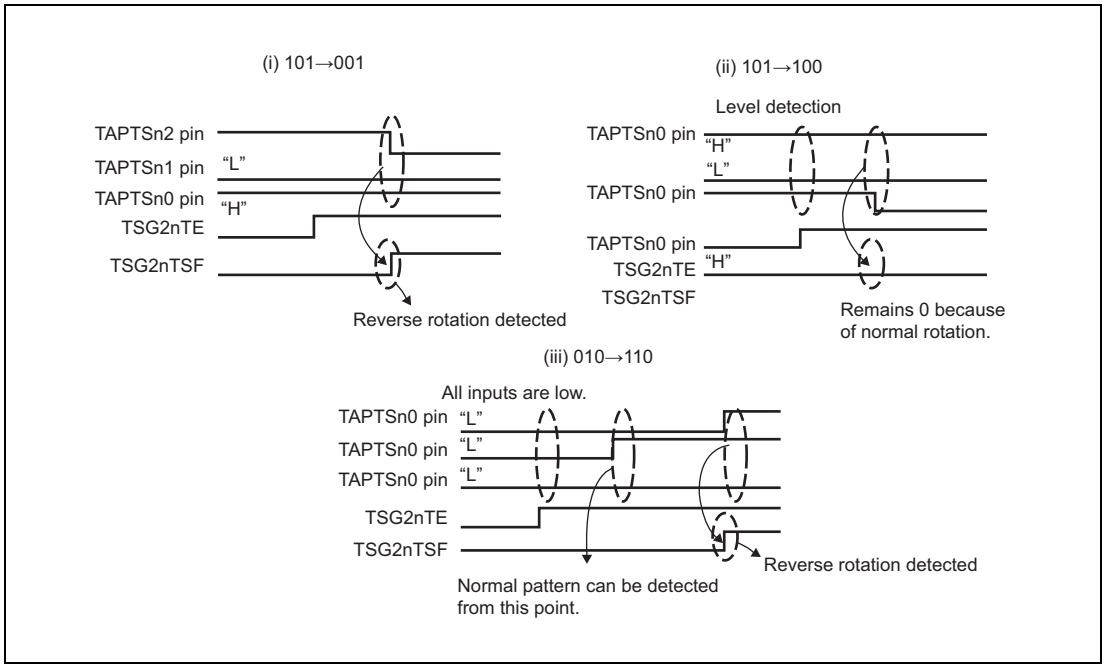


Figure 23.24 Detection of a Change (Normal/Reverse Rotation) in the Pattern Input to TAPTSn2-TAPTSn0 Pins

23.7.5.3 When Abnormal Input to TAPTSn2-TAPTSn0 Pins is Detected

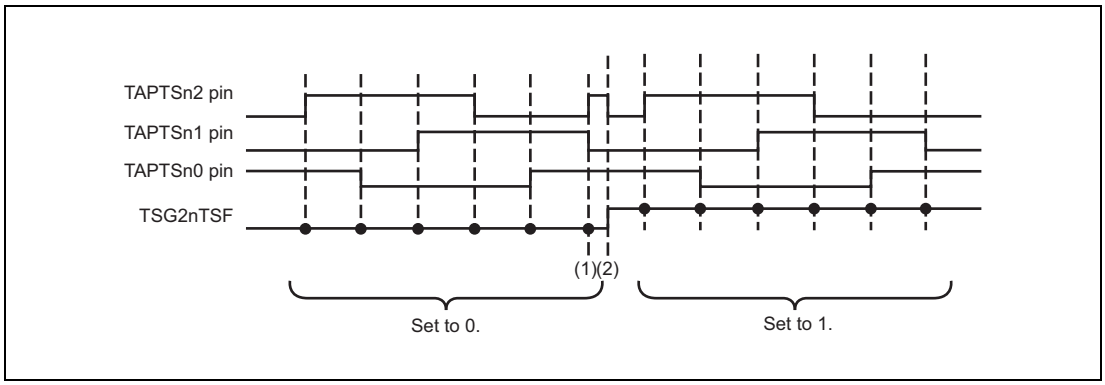


Figure 23.25 Example of Operation when Values Input to Two Pins of TAPTSn2-TAPTSn0 Change (Abnormal Operation)

(1) TSG2nTSF does not change at this point because it expects the input pattern change to {0, 1, 0} or {0, 0, 1} (if values of two pins change, TSG2nTSF does not change).

(2) The TAPTSn2-TAPTSn0 pins are judged to have been changed from {1, 0, 1} to {0, 0, 1}, and TSG2nTSF is set to 1.

Operating Mode

TSG2nTSF can be used in all operating modes.

23.7.6 Pattern Error Detection Flag (TSG2nPEF)

Name

Pattern error detection flag (TSG2nSTR2.TSG2nPEF)

Description

TSG2nPEF can detect that 000 or 111 has been input to the TAPTSn2-TAPTSn0 pins.

TSG2nPEF is set to 1 when the levels of the TAPTSn2-TAPTSn0 pins are 111 or 000, and a warning interrupt (INTTSG2nIWN) is generated. TSG2nPEF is cleared to 0 when 1 is written to TSG2nSTC.TSG2nPER.

Example of operation

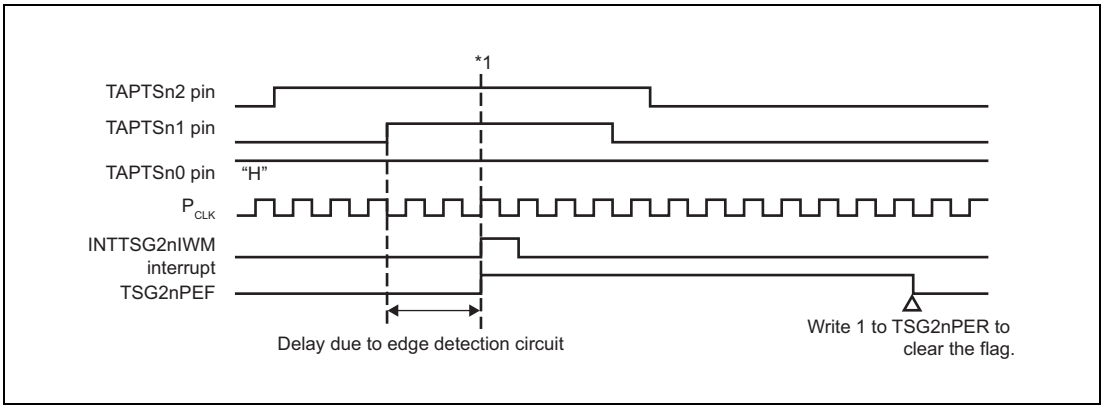


Figure 23.26 Example of Pattern Error Detection Flag Operation (TAPTSn2-TAPTSn0 Pins = 111)

CAUTION

"111" is detected.

Operating mode

TSG2nPEF can be used in all operating modes.

CAUTION

TSG2nPEF is valid only when TSG2nCTL1.TSG2nPEC = 1 and TSG2nSTR0.TSG2nTE = 1.

23.7.7 Pattern Inversion Detection Flag (TSG2nPRF)

Name

Pattern reversal inversion flag (TSG2nSTR2.TSG2nPRF)

Description

TSG2nPRF can detect that the pattern change order of the TAPTSn2 to TAPTSn0 pins have been inverted.

TSG2nPRF is set to 1 when the pattern order detection flag (TSG2nTSF) changes, and a warning interrupt (INTTSG2nIWN) is generated. However, immediately after TSG2nSTR0.TSG2nTE is set to 1, TSG2nPRF is valid at the timing of the second and subsequent change in TAPTSn2 to TAPTSn0 pins. TSG2nPRF is cleared to 0 when 1 is written to the TSG2nSTC.TSG2nPRR bit.

Example of operation

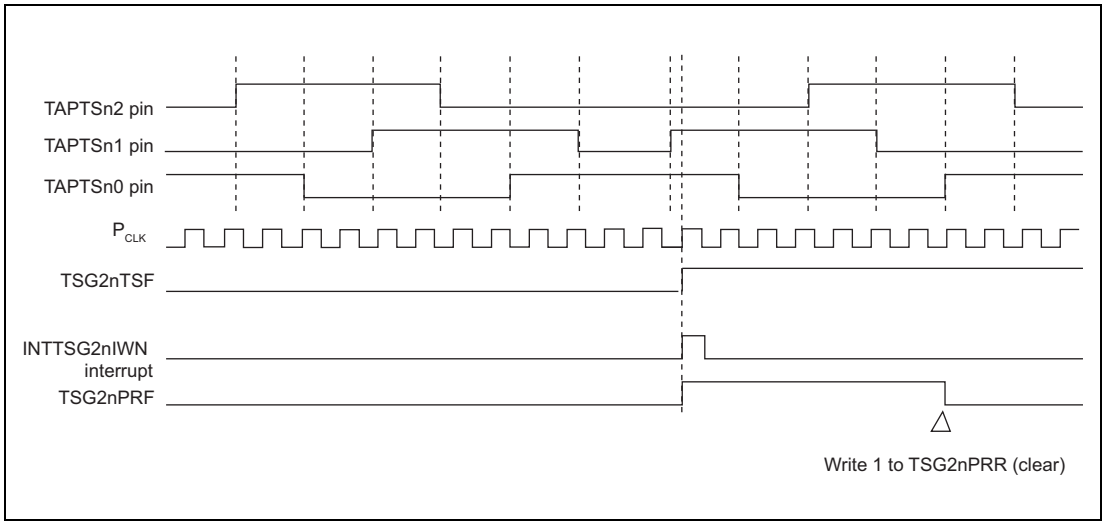


Figure 23.27 Example of Pattern Inversion Detection Flag Operation

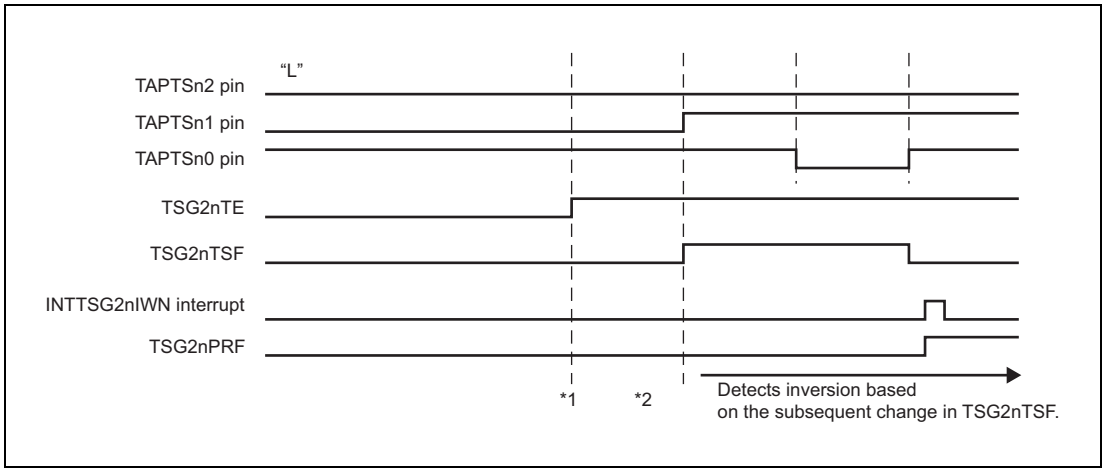


Figure 23.28 Example of Operation immediately after TSG2nTE Flag in TSG2nSTR0 is Set to 1

CAUTIONS

1. **Operation starts.**
2. **If TSG2nTSF is set to 1 by the first change in the TAPTSn2 to TAPTSn0 pins immediately after TSG2nTE is set to 1, inversion is not detected.**

Operating mode

TSG2nPEF can be used in all operating modes.

CAUTION

TSG2nPEF is valid only when TSG2nCTL1.TSG2nPEC = 1 and TSG2nSTR0.TSG2nTE = 1.

23.7.8 Pattern Phase Difference Detection Flag (TSG2nPPF)

Name

Pattern phase difference detection flag (TSG2nSTR2.TSG2nPPF)

Description

TSG2nPPF can detect the phase difference between the input pattern (TAPTSn2-TAPTSn0 pins) and the output pattern (TSG2nSTR1.TSG2nOPF2 to TSG2nOPF0 flags).

TSG2nPPF is set to 1 when the pattern phase difference is detected, and a warning interrupt (INTTSG2nIWN) is generated. TSG2nPPF remains 1 until it is cleared to 0 when 1 is written to TSG2nSTC.TSG2nPPR by software. When the phase difference is detected, TSG2nPPF is set at each operation clock cycle (PCLK). TSG2nPPF should be cleared to 0 when no phase difference occurs.

Table 23.43 Correspondence between Normal Input Patterns and Output Patterns

TAPTSn2-TAPTSn0 pins (input)	“1,0,1”	“1,0,0”	“1,1,0”	“0,1,0”	“0,1,1”	“0,0,1”
TSG2nOPF2-TSG2bOPF0 flags (output)	“0,0,1” “1,0,1” “1,0,0”	“1,0,1” “1,0,0” “1,1,0”	“1,0,0” “1,1,0” “0,1,0”	“1,1,0” “0,1,0” “0,1,1”	“0,1,0” “0,1,1” “0,0,1”	“0,1,1” “0,0,1” “1,0,1”

Example of operation

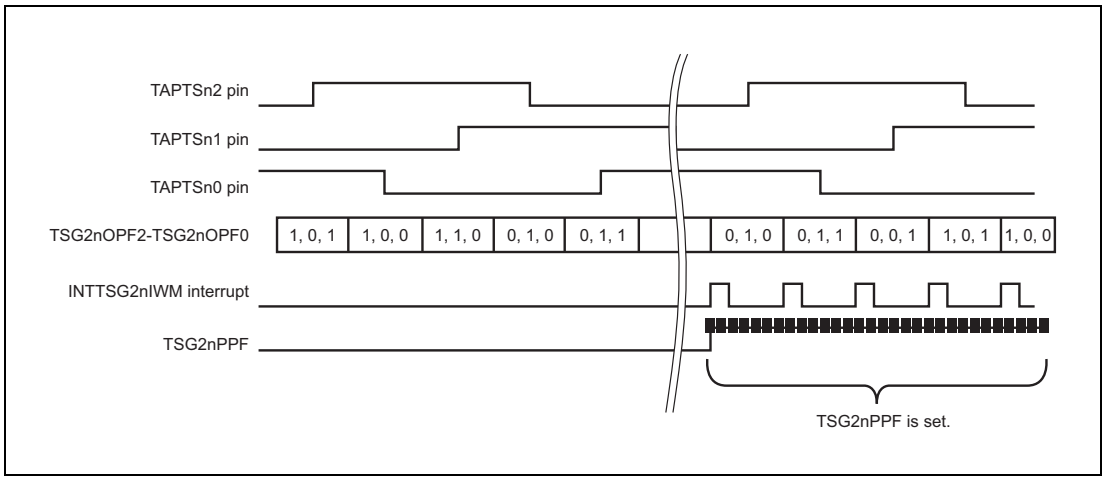


Figure 23.29 Example of Pattern Difference Detection Flag Operation

Operating mode

TSG2nPPF can be used in all operating modes.

CAUTIONS

1. **TSG2nPPF is valid only when TSG2nCTL1.TSG2nPPC = 1 and TSG2nSTR0.TSG2nTE = 1.**
2. **When 000 or 111 is input to the TAPTSn2 to TAPTSn0 pins, or when TSG2nOPF2 to TSG2nOPF0 are set to 000 or 111, TSG2nPPF is not set.**

23.7.9 Timer Output Pattern Flag (TSG2nOPF2-TSG2nOPF0)

Name

Timer output pattern flag (TSG2nSTR1.TSG2nOPF2-TSG2nOPF0)

Description

TSG2nOPF2 to TSG2nOPF0 flags indicate the timer output patterns.

Operating mode

TSG2nOPF2 to TSG2nOPF0 can be used in all operating modes.

23.7.10 Pattern Switch Detection Signal (TSG2nPTE)

Name

Pattern switch detection signal (TSG2nPTE signal)

Description

The TSG2nPTE signal toggles when the input pattern (TAPTSn2-TAPTSn0 pins) changes.

The toggle pattern is determined by the TSG2nPSC bit (TSG2nOPT0.TSG2nPSS = 1).

- TSG2nPSC = 0

		TAPTSn2-TAPTSn0 Pins after Change							
		000	111	101	100	110	010	011	001
Current TAPTSn2-TAPTSn0 pins	000	—	—	—	—	—	—	—	—
	111	—	—	—	—	—	—	—	—
	101	—	—	—	Toggle	—	—	—	—
	100	—	—	—	—	Toggle	—	—	—
	110	—	—	—	—	—	Toggle	—	—
	010	—	—	—	—	—	—	Toggle	—
	011	—	—	—	—	—	—	—	Toggle
	001	—	—	Toggle	—	—	—	—	—

- TSG2nPSC = 1

		TAPTSn2-TAPTSn0 Pins after Change							
		000	111	101	100	110	010	011	001
Current TAPTSn2-TAPTSn0 pins	000	—	—	—	—	—	—	—	—
	111	—	—	—	—	—	—	—	—
	101	—	—	—	—	—	—	—	Toggle
	100	—	—	Toggle	—	—	—	—	—
	110	—	—	—	Toggle	—	—	—	—
	010	—	—	—	—	Toggle	—	—	—
	011	—	—	—	—	—	Toggle	—	—
	001	—	—	—	—	—	—	Toggle	—

Example of operation

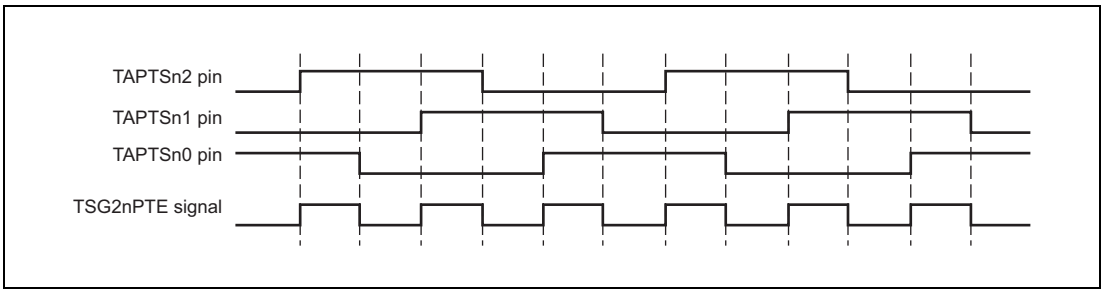


Figure 23.30 Example of Pattern Switch Detection Signal Operation

23.8 Operation of Interrupt Skipping Function

Operation related to the interrupt skipping function is described below.

Peak interrupts (INTTSG2nIPEK) and trough interrupts (INTTSG2nIVLY) can be skipped.

- TSG2nCTL4.TSG2nPIE enables output of the INTTSG2nIPEK interrupt and specifies whether to skip the interrupts.
- TSG2nCTL4.TSG2nVIE enables output of the INTTSG2nIVLY interrupt and specifies whether to skip the interrupts.

When TSG2nCTL3.TSG2nRIA is set to 1 (with reload skipping), reload is executed at the same timing as the interrupt after being skipped.

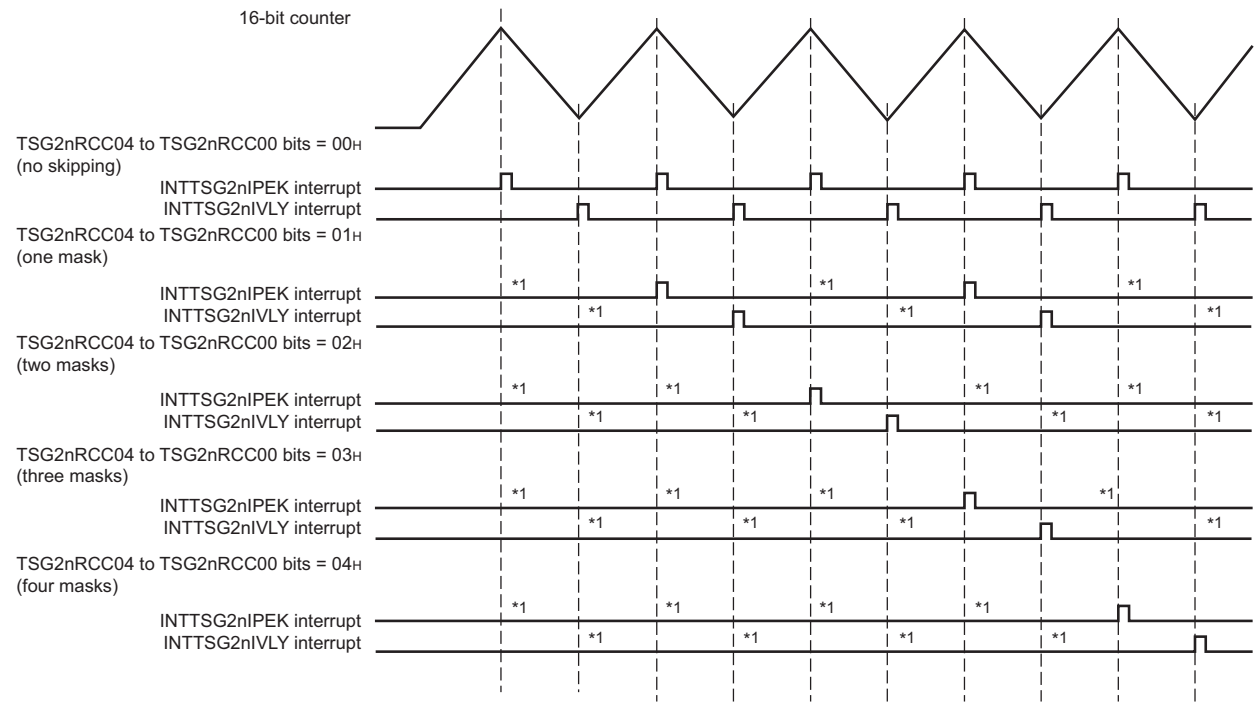
When TSG2nCTL3.TSG2nRIA is set to 0 (without reload skipping), reload is executed at the specified reload timing regardless of interrupt skipping.

CAUTION

When a value is written to TSG2nCTL4, and TSG2nRCC04 to TSG2nRCC00 are transferred to the buffer register, the interrupt skipping counter is cleared. Therefore, when the interrupt skipping function is used, the interrupt interval may be long temporarily. To avoid this, the interrupt skipping count should be changed with the reload timing being set to the interrupts skipped (TSG2nCTL3.TSG2nRIA = 1).

23.8.1 Operation of Interrupt Skipping Function

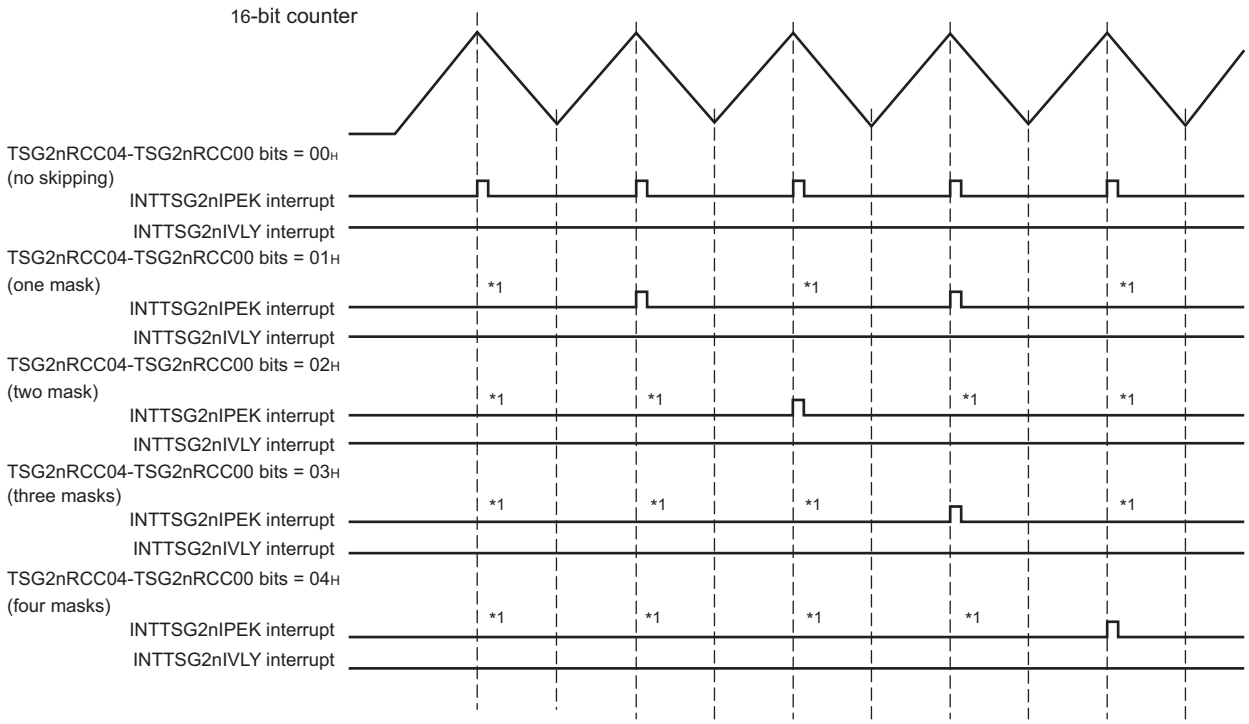
23.8.1.1 Interrupt Skipping Operation when TSG2nPIE = 1 and TSG2nVIE = 1 in TSG2nCTL4 (Peak and Trough Interrupt Generation in HT-PWM Mode)



Note 1. Skipped interrupt request

Figure 23.31 When TSG2nPIE = 1 and TSG2nVIE = 1 in TSG2nCTL4 (Peak and Trough Interrupt Generation in HT-PWM Mode)

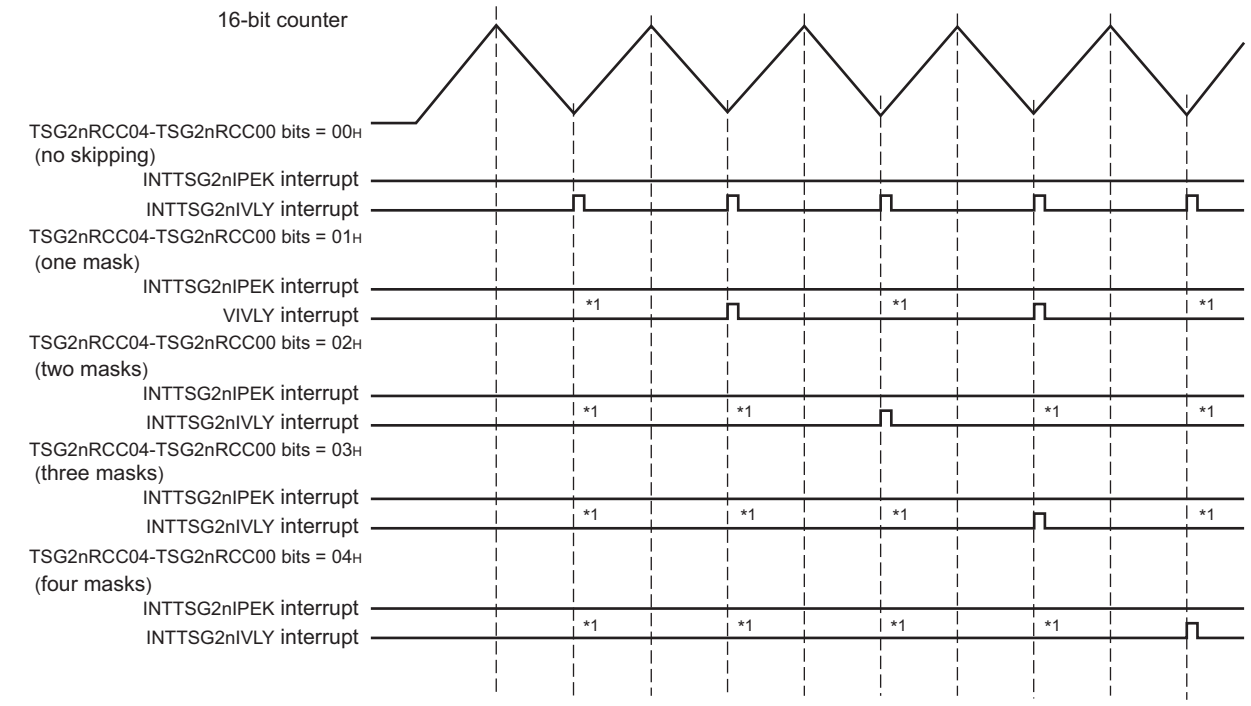
23.8.1.2 Interrupt Skipping Operation when TSG2nPIE = 1 and TSG2nVIE = 0 in TSG2nCTL4 (Generation of Only Peak Interrupts in HT-PWM Mode)



Note 1. Skipped interrupt request

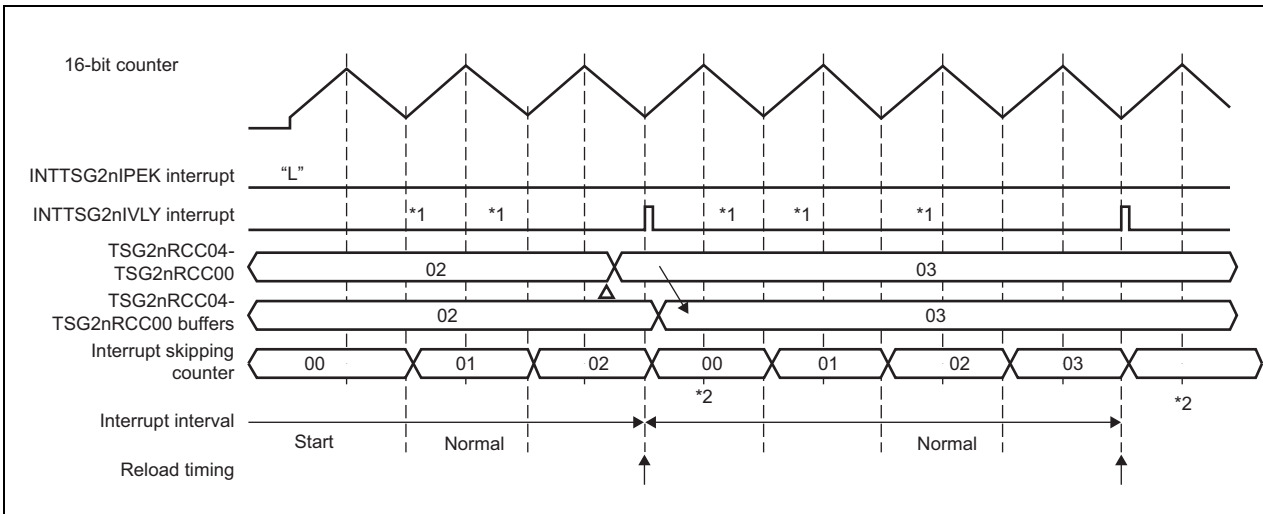
Figure 23.32 When TSG2nPIE = 1 and TSG2nVIE = 0 in TSG2nCTL4 (Generation of Only Peak Interrupts in HT-PWM Mode)

23.8.1.3 Interrupt Skipping Operation when TSG2nPIE = 0 and TSG2nVIE = 1 in TSG2nCTL4 (Generation of Only Trough Interrupts in HT-PWM Mode)



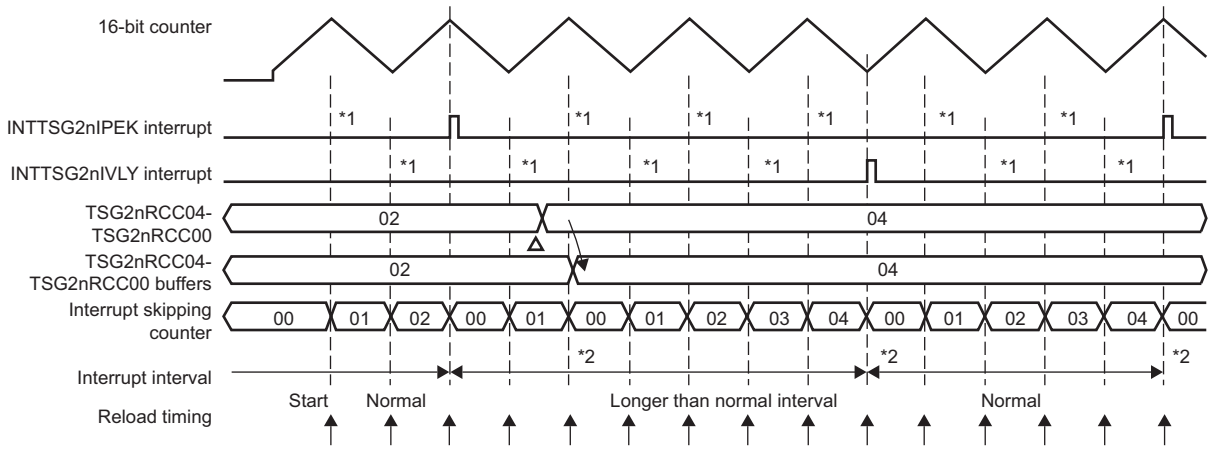
Note 1. Skipped interrupt request

Figure 23.33 When TSG2nPIE = 0 and TSG2nVIE = 1 in TSG2nCTL4 (Generation of Only Trough Interrupts in HT-PWM Mode)



Note 1. Skipped interrupt request
Note 2. Clearing of the interrupt skipping counter

Figure 23.34 When TSG2nRMC = 0 and TSG2nRIA = 1 in TSG2nCTL3 (with Reload Skipping)



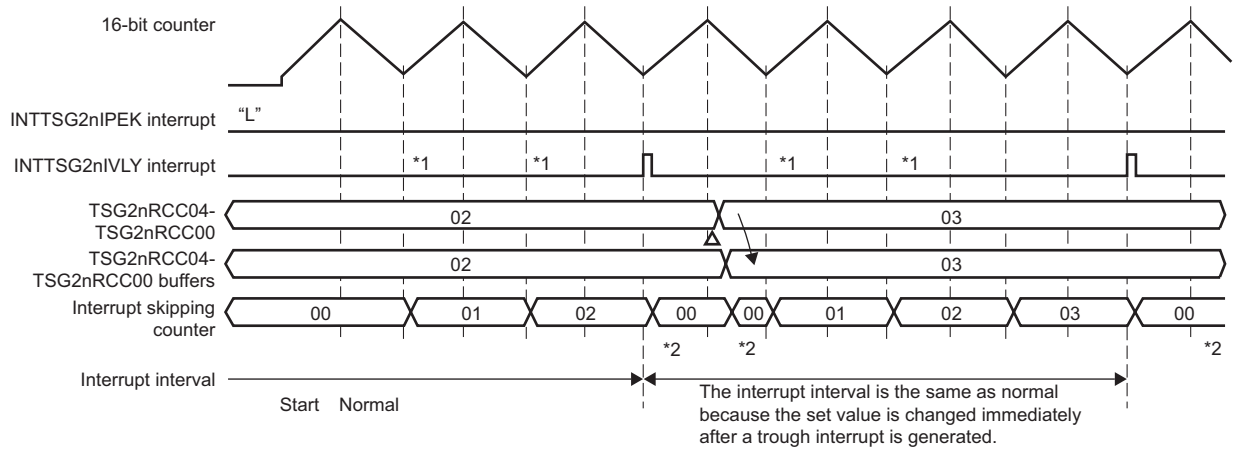
Note 1. Skipped interrupt request

Note 2. Clearing of the interrupt skipping counter

Figure 23.35 When TSG2nRMC = 0 and TSG2nRIA = 0 in TSG2nCTL3 (without Reload Skipping)

CAUTION

The interrupt interval might be longer.



Note 1. Skipped interrupt request
Note 2. Clearing of the interrupt skipping counter

Figure 23.36 When TSG2nRMC = 1 in TSG2nCTL3 (Anytime Rewriting)

CAUTION

The interrupt interval might be longer.

NOTE

After rewriting, the value is reflected immediately regardless of the reload timing. The interrupt skipping counter is cleared when the value is transferred to the TSG2nRCC04 to TSG2nRCC00 buffers, not when the pertinent register is rewritten.

23.8.2 Example of Operation when Peak Interrupt is Generated (in PWM Mode)

Operation related to the interrupt skipping function in PWM mode is described below.

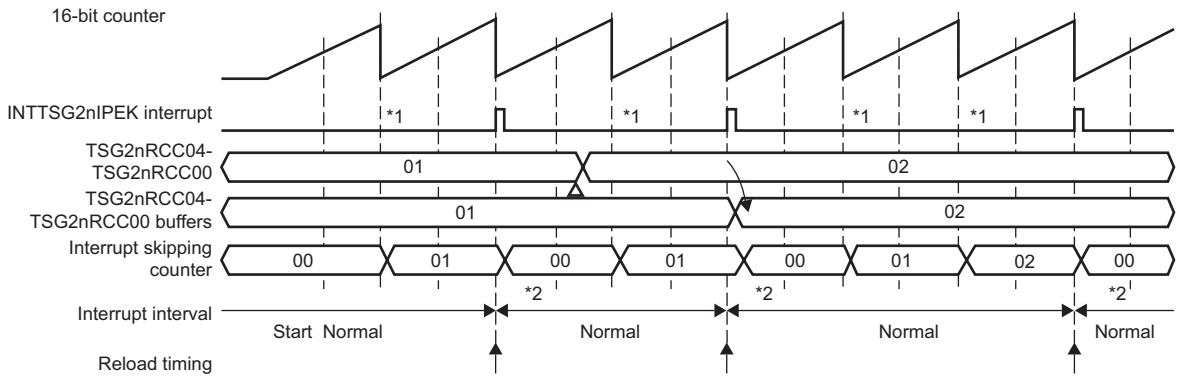
- Trough interrupts (INTTSG2nIPEK) can be skipped. In PWM mode, it is generated on compare match between the TSG2nCMP0 buffer register and the 16-bit counter.
- TSG2nCTL4.TSG2nPIE enables output of the INTTSG2nIVLY interrupt and specifies whether to skip the interrupts.
- The setting of TSG2nCTL4.TSG2nVIE is disabled. At this time, the INTTSG2nIVLY interrupt is not generated.

When TSG2nCTL3.TSG2nRIA is set to 1 (with reload skipping), reload is executed at the same timing as the interrupt after being skipped.

CAUTION

When a value is written to TSG2nCTL4, and TSG2nRCC04 to TSG2nRCC00 are transferred to the buffer register, the interrupt skipping counter is cleared. Therefore, when the interrupt skipping function is used, the interrupt interval may be long temporarily. To avoid this, the interrupt skipping count should be changed with the reload timing being set to the interrupts skipped (TSG2nCTL3.TSG2nRIA = 1).

23.8.2.1 Example of Operation



Note 1. Skipped interrupt request
Note 2. Clearing of the interrupt skipping counter

Figure 23.37 When TSG2nCTL3.TSG2nRMC = 0, TSG2nRIA = 1 and TSG2nCTL4.TSG2nPRE = 1 (Recommended Setting)

NOTE

When TSG2nCTL3.TSG2nRIA = 1, reload is executed at the same timing as the interrupt after being skipped.

23.9 A/D Conversion Trigger Function

A/D conversion trigger operation is described below.

TSG2nDCMP0W and TSG2nDCMP2 are used as compare registers of the A/D conversion trigger function.

23.9.1 A/D Operation of A/D Conversion Trigger

TSG2n has a function to generate A/D conversion start triggers (TSTADT0 and TSTADT1 signals) by selecting any of ten trigger sources as required.

The trigger sources are selected by TSG2nAT09 to TSG2nAT00 in TSG2nCTL5 and TSG2nAT19 to TSG2nAT10 in TSG2nCTL6.

23.9.1.1 TSTADT0/TSTADT1 Signal Output Control (TSG2nCTL5 and TSG2nCTL6)

[Trigger source]

- TSG2nAT00/TSG2nAT10 = 1: A trough interrupt (INTTSG2nIVLY) causes an A/D conversion trigger pulse to be generated.
- TSG2nAT01/TSG2nAT11 = 1: A peak interrupt (INTTSG2nIPEK) causes an A/D conversion trigger pulse to be generated.
- TSG2nAT02/TSG2nAT12 = 1: While the 16-bit counter is counting up, a TSG2nDCMP0 compare match enables A/D conversion trigger to be generated.
- TSG2nAT03/TSG2nAT13 = 1: While the 16-bit counter is counting down, a TSG2nDCMP0 compare match enables A/D conversion trigger to be generated.
- TSG2nAT04/TSG2nAT14 = 1: While the 16-bit counter is counting up, a TSG2nDCMP1 compare match enables A/D conversion trigger to be generated.
- TSG2nAT05/TSG2nAT15 = 1: While the 16-bit counter is counting down, a TSG2nDCMP1 compare match enables A/D conversion trigger to be generated.
- TSG2nAT06/TSG2nAT16 = 1: While the 16-bit counter is counting up, a TSG2nDCMP2 compare match enables A/D conversion trigger to be generated.
- TSG2nAT07/TSG2nAT17 = 1: While the 16-bit counter is counting down, a TSG2nDCMP2 compare match enables A/D conversion trigger to be generated.
- TSG2nAT08/TSG2nAT18 = 1: A trough timing of the 16-bit sub-counter (at a switch from decrementing to incrementing) enables A/D conversion trigger to be generated.
- TSG2nAT09/TSG2nAT19 = 1: A peak timing of the 16-bit sub-counter (at a switch from incrementing to decrementing) enables A/D conversion trigger to be generated.

[Skipping setting]

- TSG2nACC01 and TSG2nACC00, and TSG2nACC11 and TSG2nACC10:Set the skipping rate of the TSTADT0 and TSTADT1 signals, respectively.

All A/D conversion triggers selected by TSG2nAT09 to TSG2nAT00 and TSG2nAT19 to TSG2nAT10 are logically ORed, and the resultant signals are subjected to skipping control specified by TSG2nACC01 and TSG2nACC00, and TSG2nACC11 and TSG2nACC10, and then the TSTADT0 and TSTADT1 signals are generated.

A peak interrupt (INTTSG2nIPEK) and a trough interrupt (INTTSG2nIVLY) selected by TSG2nAT00 and TSG2nAT01, and TSG2nAT10 and TSG2nAT11 are interrupt signals obtained after skipped. Therefore, they are output at the timing according to interrupt skipping control. If the interrupt output is not enabled by TSG2nCTL4.TSG2nPIE and TSG2nVIE, A/D conversion trigger is not output.

TSG2nACC01, TSG2nACC00, and TSG2nAT09 to TSG2nAT00, and TSG2nACC11, TSG2nACC10, and TSG2nAT19 to TSG2nAT10 can be rewritten during timer operation.

If A/D conversion trigger setting bits are rewritten during operation, the rewritten values are reflected on the A/D conversion trigger output status immediately. Such control bits are rewritten at anytime regardless of operating modes. If a write access is made to TSG2nCTL5 and TSG2nCTL6 (including a rewrite of the same value), the A/D conversion trigger skipping counter is cleared and starts counting from zero.

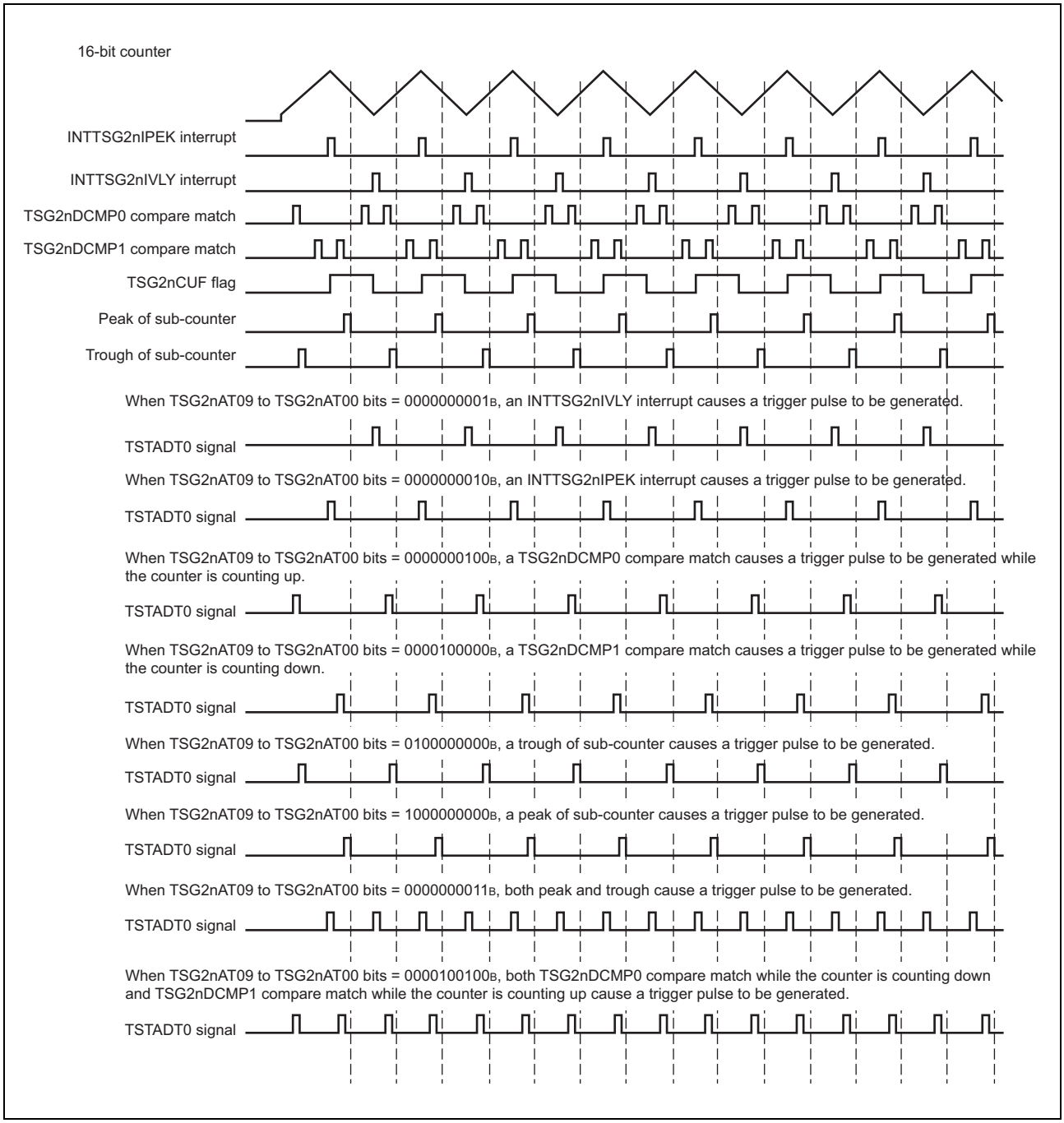


Figure 23.38 When TSG2nPIE = 1, TSG2nVIE = 1, and TSG2nRCC04 to TSG2nRCC00 = 00_B in TSG2nCTL4, and TSG2nACC01 and TSG2nACC00 = 00_B in TSG2nCTL5 (HT-PWM Mode)

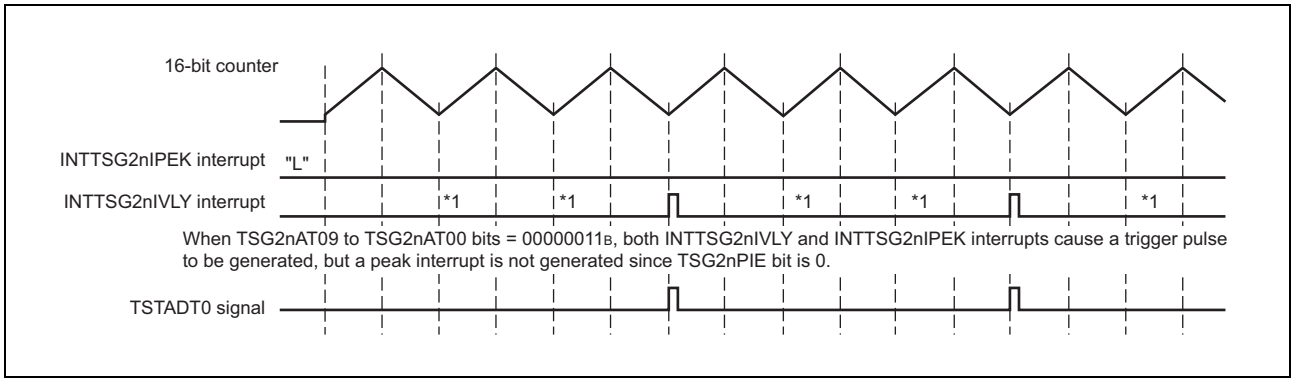


Figure 23.39 When TSG2nPIE = 0, TSG2nVIE = 1, and TSG2nRCC04 to TSG2nRCC00 = 02_B in TSG2nCTL4 and TSG2nACC01 and TSG2nACC00 = 00_B in TSG2nCTL5 (HT-PWM Mode)

Note1. Skipped interrupt request

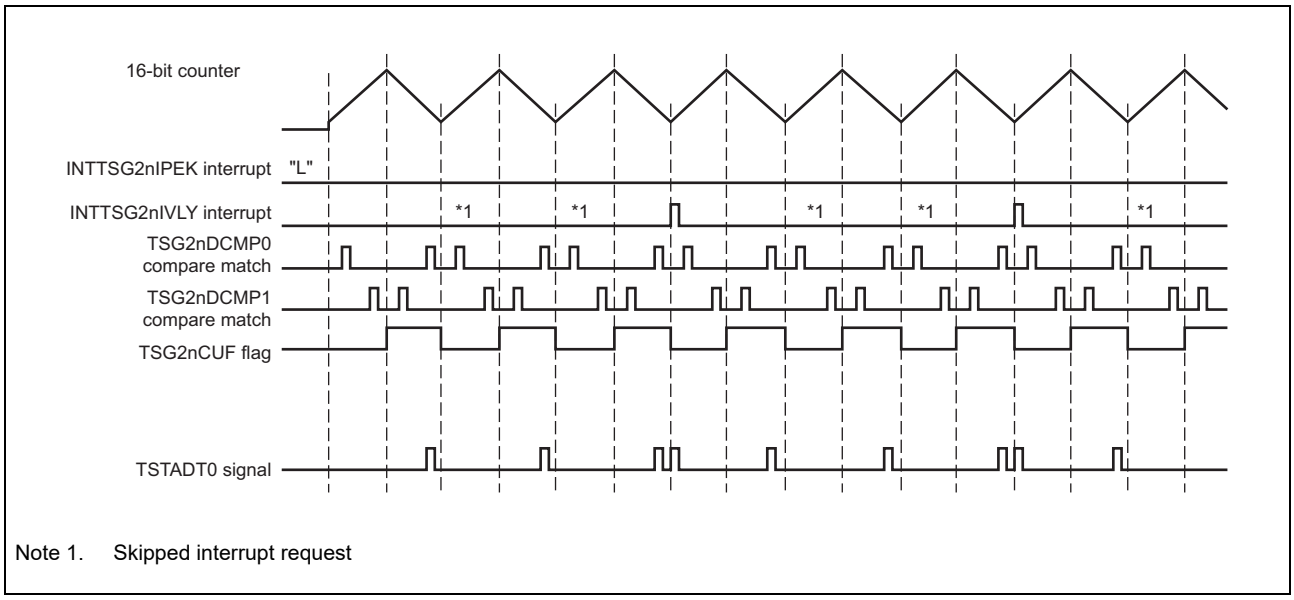


Figure 23.40 When TSG2nPIE = 0, TSG2nVIE = 1, and TSG2nRCC04 to TSG2nRCC00 = 02_B in TSG2nCTL4 and TSG2nACC01 and TSG2nACC00 = 00_B, and TSG2nAT09 to TSG2nAT00 = 0000 1001_B in TSG2nCTL5 (HT-PWM Mode)

23.9.1.2 A/D Conversion Trigger Skipping Function

Example of operation of the A/D conversion trigger skipping function is shown in Figure below.

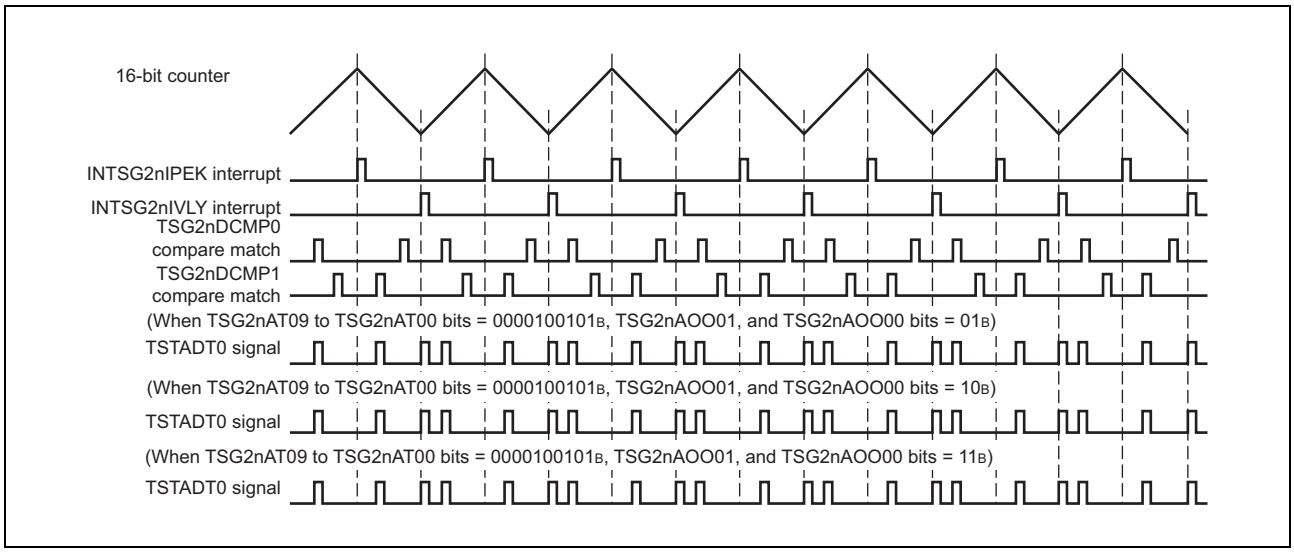


Figure 23.41 Example of Operation of A/D Conversion Trigger Skipping Function

NOTE

Broken-lined pulses indicate A/D conversion trigger pulses skipped by the A/D conversion trigger skipping function.

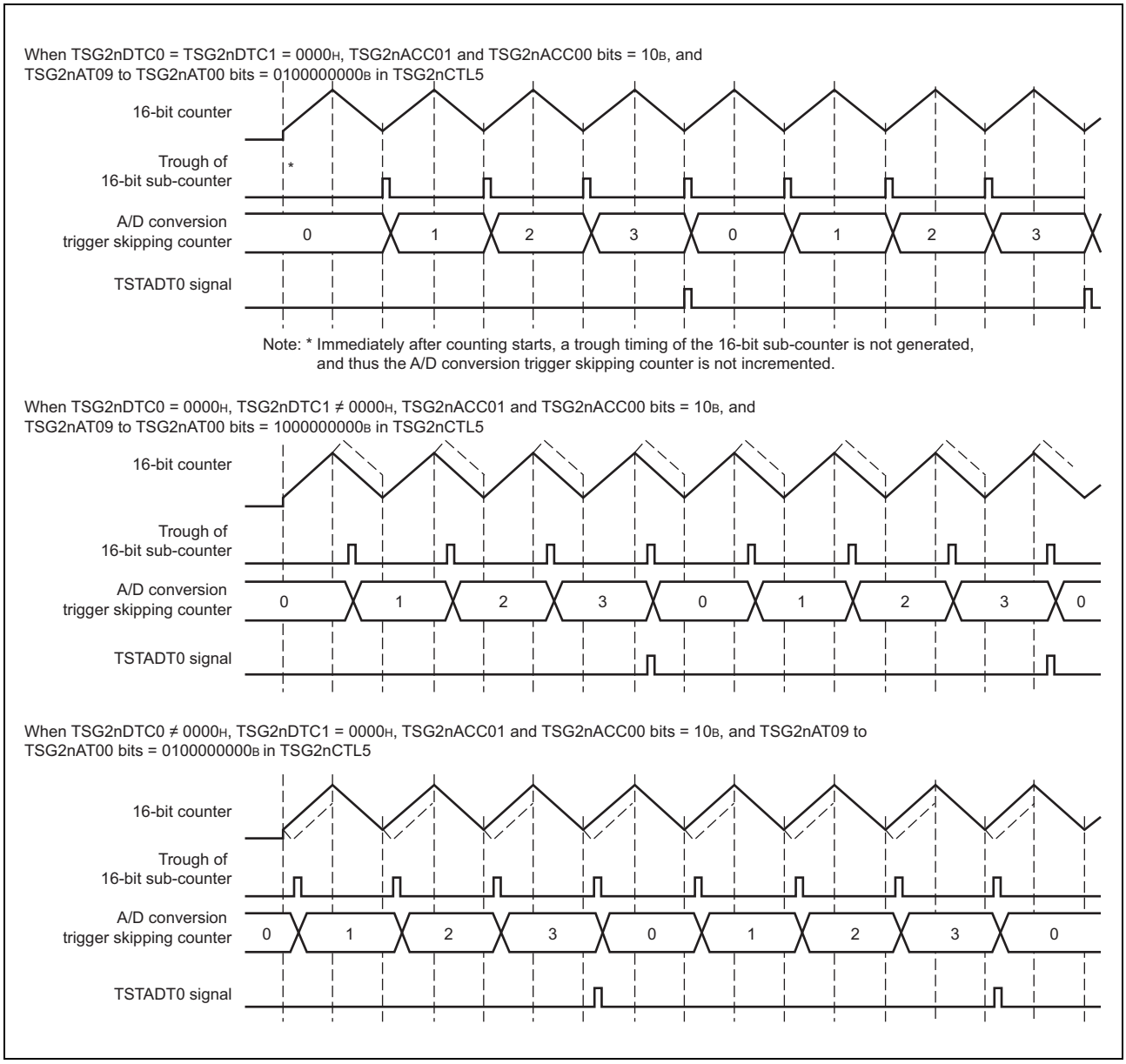


Figure 23.42 Example of Operation of A/D Conversion Trigger Skipping Function

23.9.1.3 Notes on A/D Conversion Trigger

- If the same value is written to TSG2nDCMP0 and TSG2nDCMP1 or TSG2nDCMP2, and the same condition (when the 16-bit counter increments or decrements) is set as the valid A/D conversion trigger, A/D conversion trigger skipping counter is incremented by one and one trigger pulse is output upon a match of the 16-bit counter with these registers.
- In PWM mode, SP-PWM mode, and 120-DC mode, a trough interrupt (INTTSG2nIVLY) is not generated. Only a peak interrupt (INTTSG2nIPEK) is valid.
- In 120-DC mode, when the 16-bit counter is cleared during the carrier period due to switch of the output pattern, the A/D conversion trigger is not generated if TSG2nDCMP2 to TSG2nDCMP0 values do not match with the 16-bit counter value and a peak interrupt (INTTSG2nIPEK) is not generated.

23.10 Error and Warning Interrupts

23.10.1 Error Interrupt Function

If simultaneous activation of the positive and negative phases is detected after the error interrupt function is enabled (TSG2nIOC1.TSG2nEOC = 1), TSG2nSTR2.TSG2nTBF is set, and an error interrupt (INTTSG2nIER) of TSG2n is generated. Whether or not to detect an error of each phase (TSOn1 and TSOn2, TSOn3 and TSOn4, and TSOn5 and TSOn6 pins) can be selected by TSG2nCTL1.TSG2nTBA2 to TSG2nTBA0, respectively.

When an error occurs, outputs of the TSOn1 to TSOn6 pins can be set to high-impedance.

The following table shows whether or not simultaneous activation of the positive and negative phases can be detected in each mode.

Mode	Positive Phase and Negative Phase Simultaneous Active State Detection
PWM mode	Possible
HT-PWM mode	Possible
SP-PWM mode	Possible
120-DC mode	Possible

CAUTION

When an error interrupt is generated, the error status should be canceled within an error interrupt handling. Otherwise, subsequent error interrupts are not generated.

23.10.1.1 PWM Mode and 120-DC Mode

In PWM mode, if TSG2nCMP1 and TSG2nCMP2, and TSG2nCMP3 and TSG2nCMP4 are set so that the TSOn1 and TSOn2 pins output the active level simultaneously, an error interrupt (INTTSG2nIER) is generated. With the same setting, if TSG2nCMP5, TSG2nCMP6, TSG2nCMP7, TSG2nCMP8, TSG2nCMP9, TSG2nCMP10, TSG2nCMP11, and TSG2nCMP12 are set so that the TSOn3 and TSOn4, and TSOn5 and TSOn6 pins output the active level simultaneously, an error interrupt (INTTSG2nIER) is generated.

In 120-DC mode, if TSG2nCMP1, TSG2nCMP2, TSG2nCMP5, TSG2nCMP6, TSG2nCMP9, and TSG2nCMP10, TSG2nCMP3, TSG2nCMP4, TSG2nCMP7, TSG2nCMP8, TSG2nCMP11, and TSG2nCMP12, and TSG2nPAT0W and TSG2nPAT1W are set so that the TSOn1 and TSOn2 pins output the active level simultaneously, an error interrupt (INTTSG2nIER) is generated. With the same setting, the TSOn3 and TSOn4, and TSOn5 and TSOn6 pins also output the active level simultaneously, an error interrupt (INTTSG2nIER) is generated.

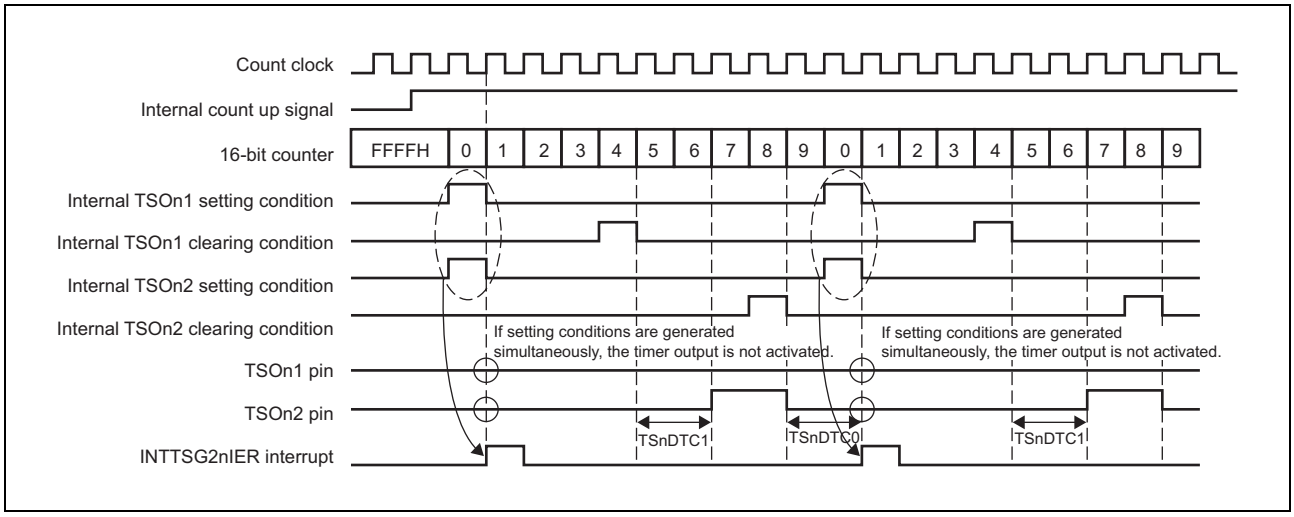


Figure 23.43 Example of Error Interrupt (INTTSG2nIER) Generation (PWM Mode)

NOTE

TSOn3 and TSOn4, and TSOn5 and TSOn6 behave the same.

When the active level of output is switched by operating TSG2nIOC2.TSG2nOL1 and TSG2nOL2, an error interrupt is generated as shown in the figure below.

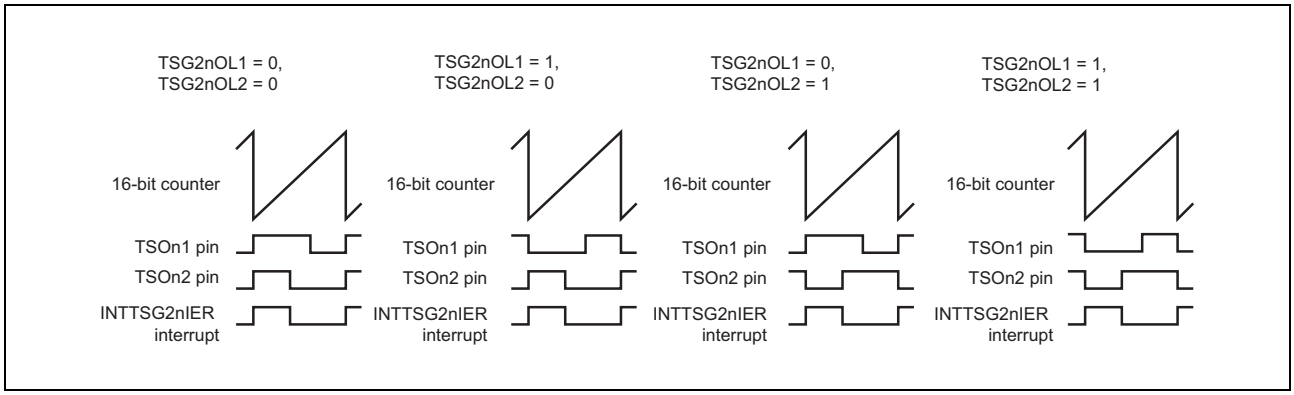


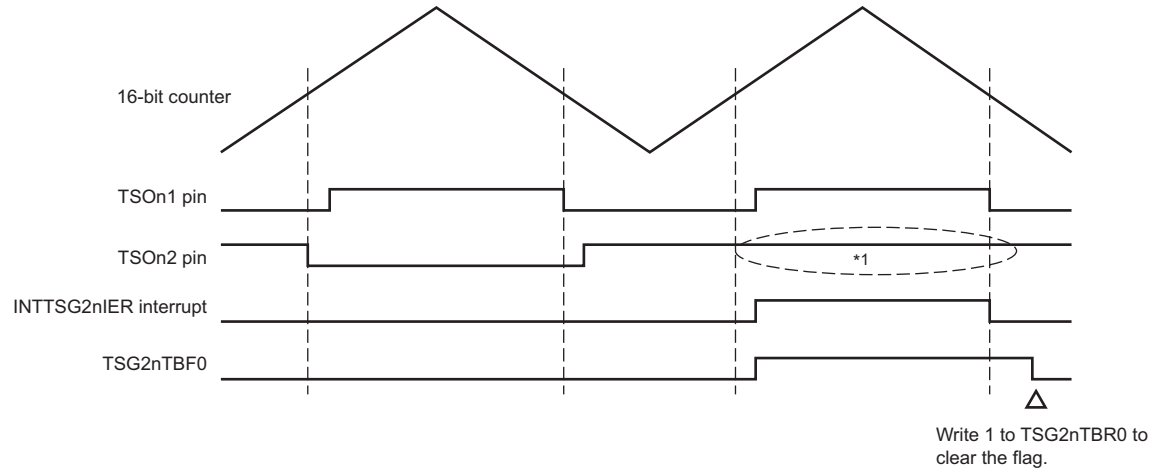
Figure 23.44 Example of Error Interrupt (INTTSG2nIER) Generation for Each Active Level

23.10.1.2 HT-PWM Mode and SP-PWM Mode

Either TSG2n dead time setting register 0 or 1 (TSG2nDTC0 or TSG2nDTC1) is 0000_H, an error may occur.

NOTE

If an error occurs when the dead time control function is used (both TSG2nDTC0 and TSG2nDTC1 are not 0000_H), internal circuit failure may occur.



Note 1. TSOn2 pin control circuit failure occurs.

Figure 23.45 Example of Error Interrupt Operation

23.10.2 Warning Interrupt Function

TSG2n has a warning interrupt (INTTSG2nIWN).

Warning interrupt (INTTSG2nIWN) is generated when any of the following conditions is detected.

For details, see **Section 23.7, Flags**.

- When simultaneous change in two or more pins of TAPTSn2-TAPTSn0 is detected
  See **Section 23.7.4, Noise Detection Flag (TSG2nNDF)**.
- When inversion is detected on the TAPTSn2-TAPTSn0
  See **Section 23.7.7, Pattern Inversion Detection Flag (TSG2nPRF)**.
- When 000 or 111 is detected from the TAPTSn2-TAPTSn0 pins
  See **Section 23.7.6, Pattern Error Detection Flag (TSG2nPEF)**.
- When the phase difference between the input pattern (TAPTSn2 to TAPTSn0 pins) and output patterns (TSG2nOPF2 to TSG2nOPF0) is detected
  See **Section 23.7.8, Pattern Phase Difference Detection Flag (TSG2nPPF)**.

23.11 Operating Modes

Table 23.44 List of Modes

TSG2nCTL0 Register		
TSG2nMD1	TSG2nMD0	Timer Modes
0	0	PWM mode
0	1	HT-PWM mode (HT-PWM)
1	0	Shifted-pulse - pulse width modulation mode (SP-PWM)
1	1	120-DC mode

23.11.1 PWM Mode

Overview

A PWM signal is output at the TSOn1 to TSOn6 pins according to set timing/clear timing of TSG2nCMP1 to TSG2nCMP12 registers with the PWM period set in the TSG2nCMP0 register.

Prerequisites

- Set the set timing to the compare register with an even number:
  TSG2nCMP2 (set timing of the TSOn1 output), TSG2nCMP4 (set timing of the TSOn2 output), TSG2nCMP6 (set timing of the TSOn3 output), TSG2nCMP8 (set timing of the TSOn4 output), TSG2nCMP10 (set timing of the TSOn5 output) and TSG2nCMP12 (set timing of the TSOn6 output)
- Set the clear timing to the compare register with an odd number:
  TSG2nCMP1 (clear timing of the TSOn1 output), TSG2nCMP3 (clear timing of the TSOn2 output), TSG2nCMP5 (clear timing of the TSOn3 output), TSG2nCMP7 (clear timing of the TSOn4 output), TSG2nCMP9 (clear timing of the TSOn5 output) and TSG2nCMP11 (clear timing of the TSOn6 output)

Functional description

Set the PWM period and set/clear timing of the TSOn1 to TSOn6 outputs. Set TSG2nTRG0.TSG2nTS = 1 to start the timer counter.

The TSOn1 to TSOn6 outputs are set to the inactive state at the same time the counting begins. The outputs are set to the active state by the match of the buffer registers TSG2nCMP2, TSG2nCMP4, TSG2nCMP6, TSG2nCMP8, TSG2nCMP10, and TSG2nCMP12 with the 16-bit counter.

Next, the TSOn1 to TSOn6 outputs are set to the inactive state by the match of the buffer registers TSG2nCMP1, TSG2nCMP3, TSG2nCMP5, TSG2nCMP7, TSG2nCMP9, and TSG2nCMP11 with the 16-bit counter.

During counting, a compare match interrupt (INTTSG2nI0 to INTTSG2nI12) is generated by the match of the buffer register TSG2nCMP0 to TSG2nCMP12 with the 16-bit counter.

CAUTION

Reload is executed when writing to the TSG2nCMP1 register at TSG2nCTL3.TSG2nRMC = 0. Therefore, even when it is needed to rewrite only the value of the TSG2nCMP0 register, a write operation to the TSG2nCMP1 register is necessary. When only the TSG2nCMP0 register is rewritten, reload is not done.

NOTE

The PWM mode is set when TSG2nCTL0.TSG2nMD1 and TSG2nMD0 = 00_B.

(1) When TSG2nCMP0 and TSG2nCMP1 to TSG2nCMP12 are Not Rewritten during Timer Operation

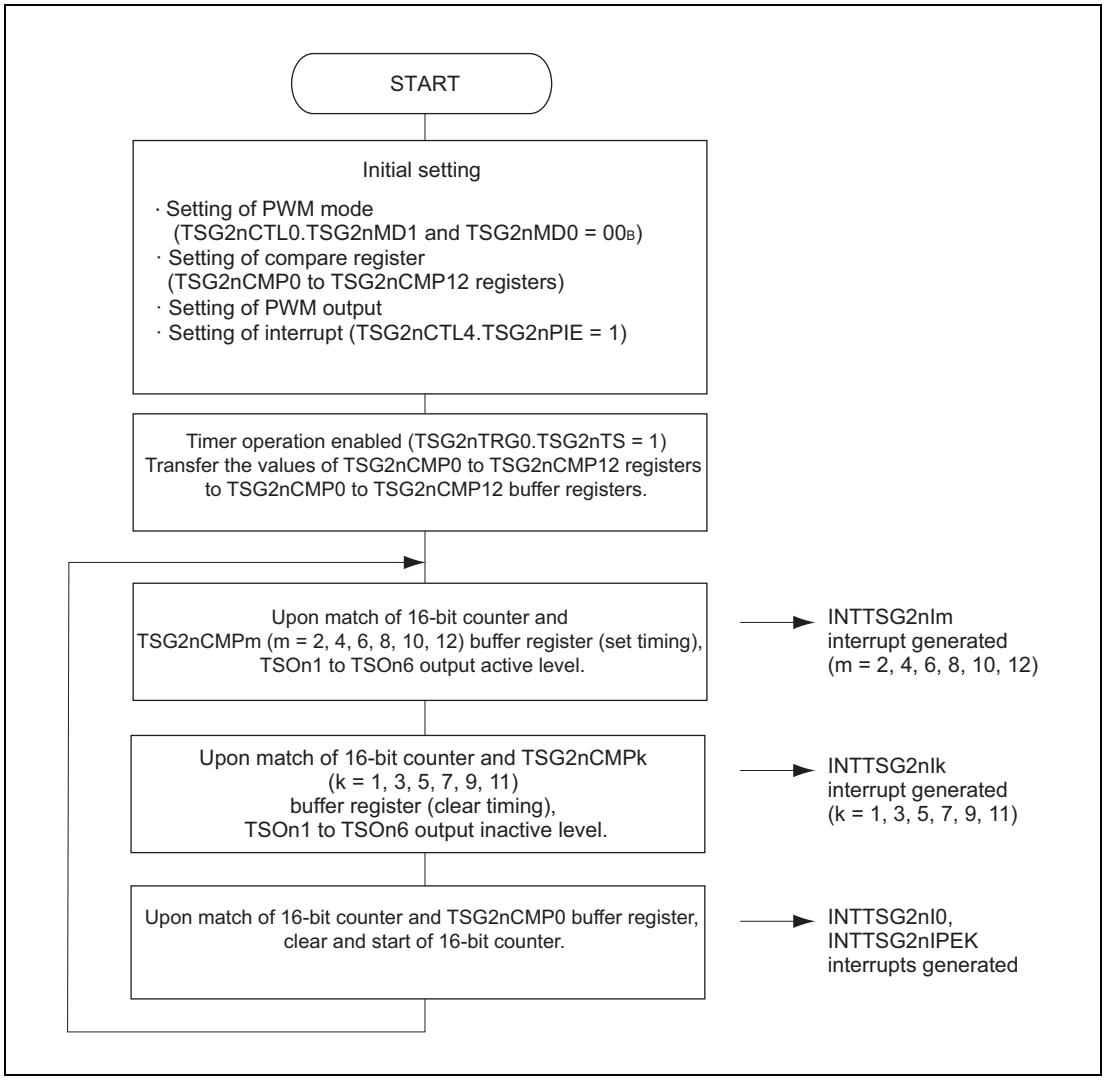
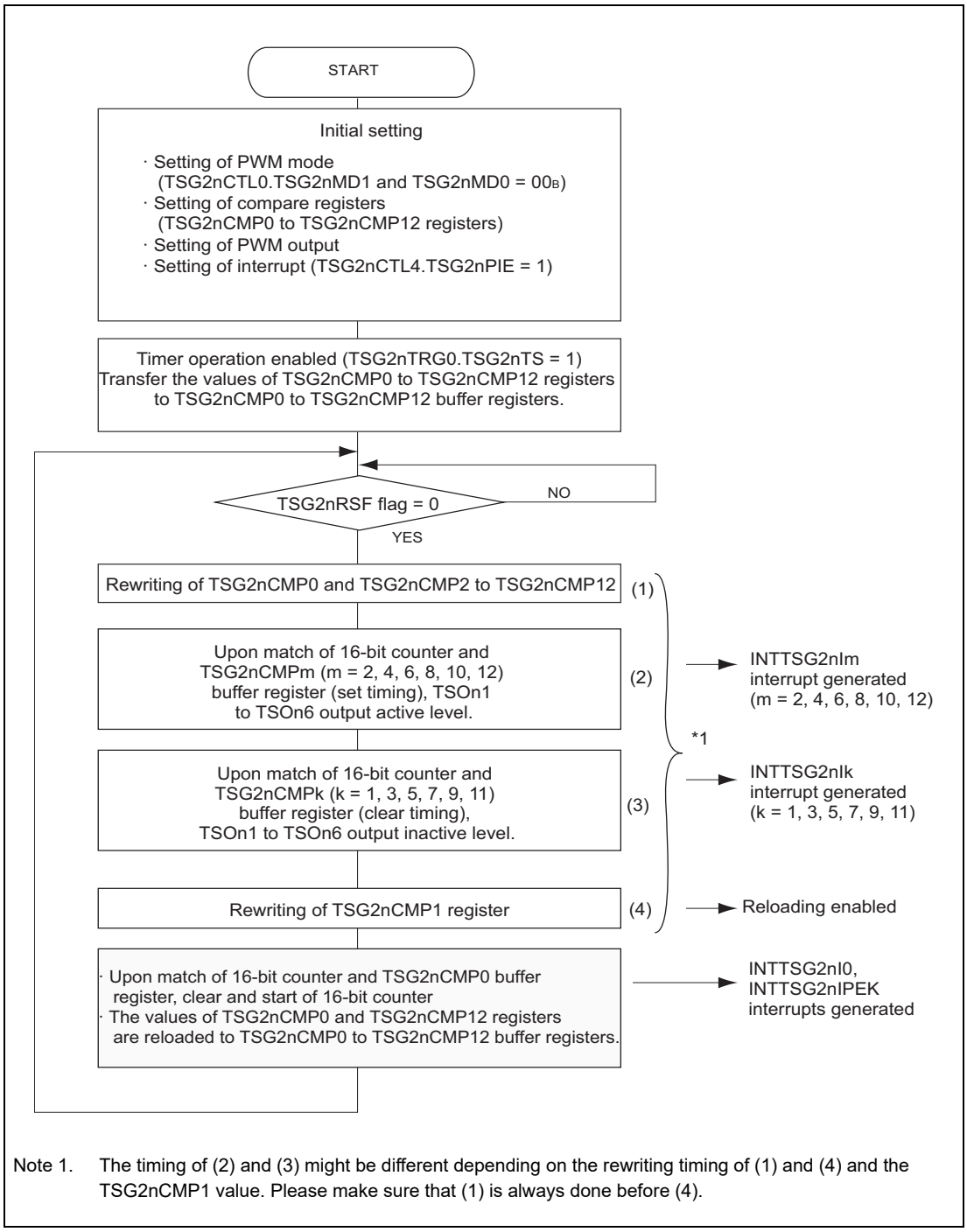


Figure 23.46 Basic Operation Flow of PWM Mode (1/2)

(2) When TSG2nCMP0 and TSG2nCMP1 to TSG2nCMP12 are Rewritten during Timer Operation



Note 1. The timing of (2) and (3) might be different depending on the rewriting timing of (1) and (4) and the TSG2nCMP1 value. Please make sure that (1) is always done before (4).

Figure 23.47 Basic Operation Flow of PWM Mode (2/2)

CAUTION

Please rewrite compare registers after confirming that the reload request flag TSG2nRSF is 0.

23.11.1.1 List of Operations in PWM Mode

Table 23.45 Counter Functions in PWM Mode

Operation		Setting Conditions
16-bit counter	Start	TSG2nTRG0.TSG2nTS = 0 → 1, or a simultaneous start trigger
	Clear	Compare match of TSG2nCMP0 buffer register and 16-bit counter
	Stop	TSG2nTRG1.TSG2nTT = 0 → 1

Table 23.46 Functions of Compare Registers and Dead Time Setting Register in PWM Mode

Register	Rewriting Method	Rewrite during Operation	Function
TSG2nCMP0	Reload/Anytime rewrite	Possible	Setting period
TSG2nCMPm (m = 1 to 12)	Reload/Anytime rewrite	Possible	Setting set/clear timing
TSG2nDCMP0W, TSG2nDCMP2	Reload/Anytime rewrite	Possible	Diagnostic output or A/D conversion trigger
TSG2nDTC0, TSG2nDTC1	Reload	Possible*1	Setting dead time

Note 1. Please refer to **Section 23.11.1.3, Controlling Dead Time in PWM Mode**.

Table 23.47 Timer Output in PWM Mode

Pin	Function
TSOnm (m = 1 to 6)	PWM output by compare match of TSG2nCMPk buffer register and 16-bit counter (k = 1 to 12)
TSOn7	Diagnostic signal output or pulse output by A/D conversion trigger

Table 23.48 Timer Output in PWM Mode

Interrupt	Function
INTTSG2nIm (m = 0 to 12)	Compare match of TSG2nCMPm buffer register and 16-bit counter (m = 0 to 12)
INTTSG2nIER	Error (simultaneous active state of TSOn1 and TSOn2, or TSOn3 and TSOn4, or TSOn5 and TSOn6)
INTTSG2nIVLY	—
INTTSG2nIPEK	Peak interrupt (generated at the same timing as INTTSG2nI0)
INTTSG2nIWN	Warning interrupt

Note: "—" : Not available in PWM mode

Table 23.49 Compare Match Timing in PWM Mode

Compare Match	Timing
TSG2nCMP0	When 16-bit counter changes from TSG2nCMP0 to 0000_H
TSG2nCMPm (m = 1 to 12)	After detecting the match of 16-bit counter and TSG2nCMPm (m = 1 to 12)

Table 23.50 Example of Setting Each Timer Output Condition in PWM Mode

Pin	Item	Output Period	Output Duty	
			Output Condition	Setting Condition
TSOnm (m = 1 ot 6)	PWM output	(TSG2nCMP0 + 1) × count clock	Output an inactive level throughout one period (duty cycle 0%)	TSG2nCMPm = TSG2nCMP (m + 1) or TSG2nCMP (m + 1) > TSG2nCMP0 (m = 1, 3, 5, 7, 9, 11)
			Output an active level of one count clock in one period	TSG2nCMPm = TSG2nCMP (m + 1) + 1 TSG2nCMP (m + 1) = TSG2nCMPm - 1 (m = 1, 3, 5, 7, 9, 11)
			Output an inactive level of one count clock in one period	TSG2nCMPm = TSG2nCMP (m + 1) - 1 TSG2nCMP (m + 1) = TSG2nCMPm + 1 (m = 1, 3, 5, 7, 9, 11)
			Output an active level throughout one period (duty cycle 100%)	TSG2nCMPm > TSG2nCMP0 TSG2nCMP (m + 1) ≤ TSG2nCMP0 (m = 1, 3, 5, 7, 9, 11)

When only the TSG2nCMP2 register is rewritten and the TSOn1 pin output is used (TSG2nIOC0.TSG2nTOE1 = 1, TSG2nIOC2.TSG2nOL1 = 0)

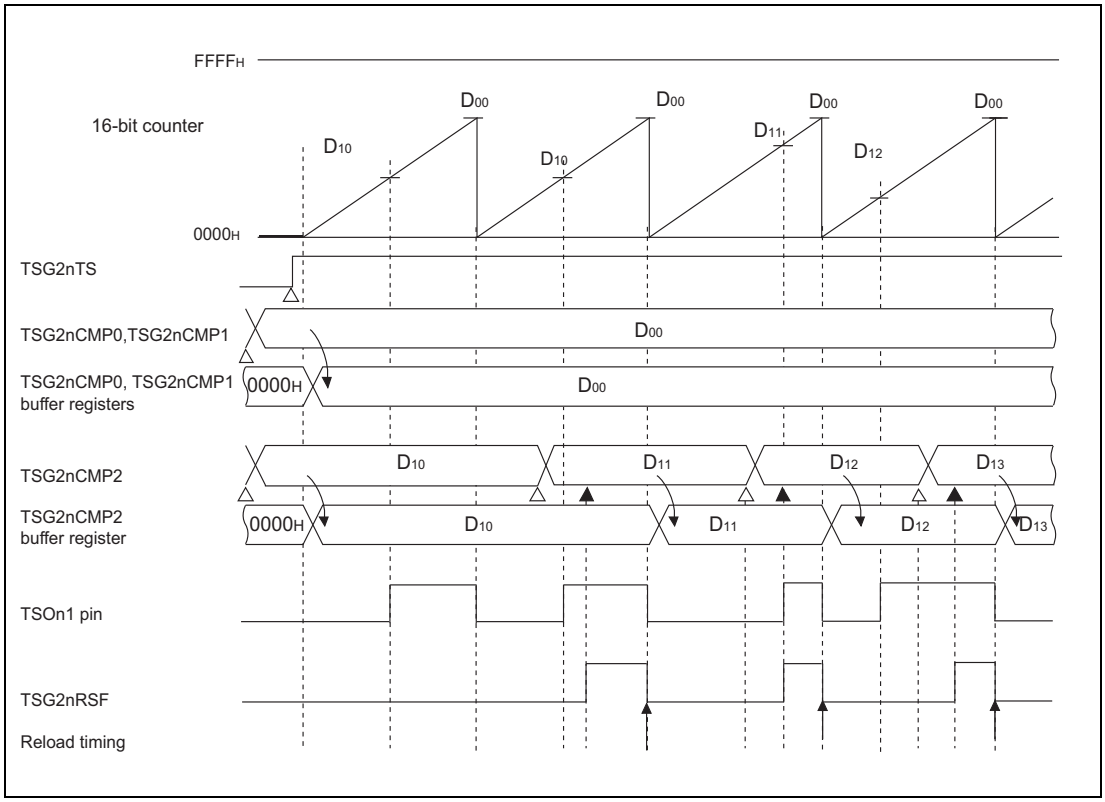
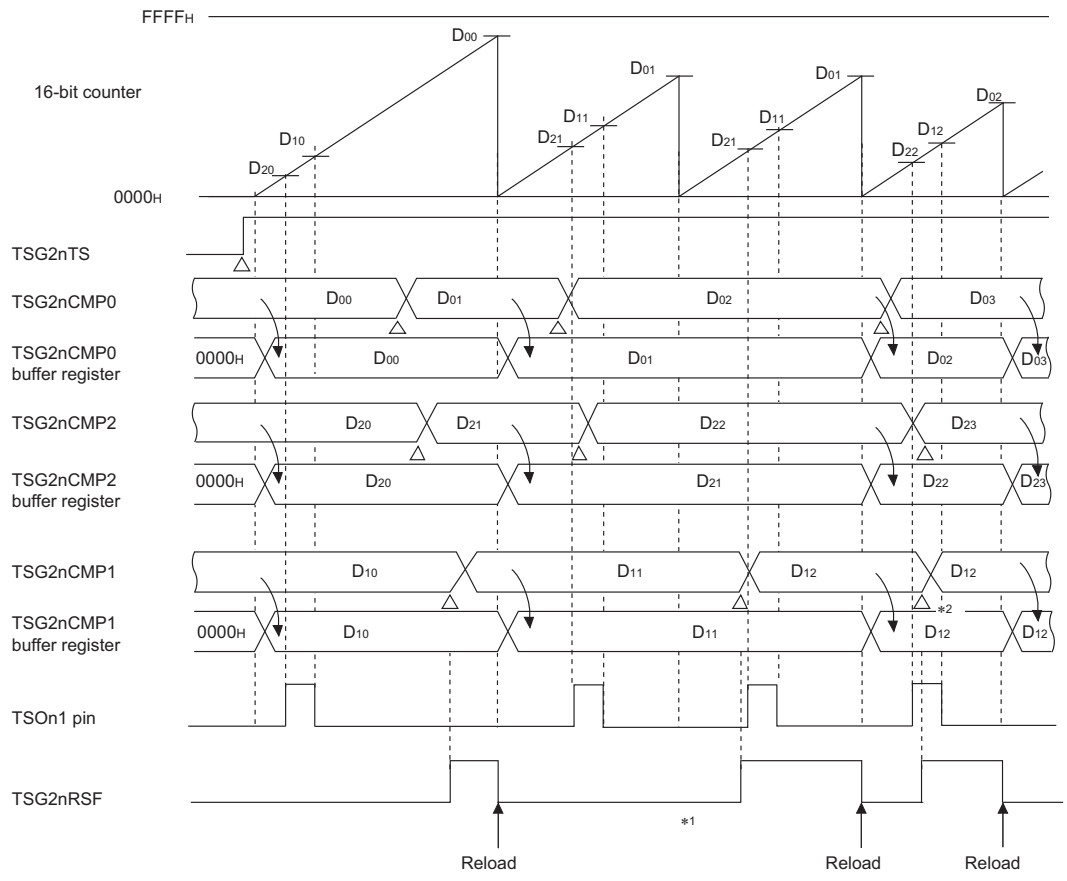


Figure 23.48 Example of Basic Operation Timing in PWM Mode (1/2)

NOTES

1. D00: Set values of TSG2nCMP0 and TSG2nCMP1 (0000_H - $FFFF_H$)
   D10, D11, D12 and D13: Set values of TSG2nCMP2 (0000_H - $FFFF_H$)
2. TSOn1 (PWM duty cycle) = (TSG2nCMP1 - TSG2nCMP2) × count clock
   TSOn1 (PWM period) = (TSG2nCMP0 + 1) × count clock
3. TSOn2 to TSOn6 pins behave similarly to the TSOn1 pin.
4. Δ: Write access
5. ▲: TSG2nCMP1 write access (equivalent).

When the TSG2nCMP0 to TSG2nCMP2 registers are rewritten, and the TSOn1 pin output is used (TSG2nIOC0.TSG2nTOE1 = 1, TSG2nIOC2.TSG2nOL1 = 0)



Note 1. The TSG2nCMP0 buffer register is not reloaded, because the TSG2nCMP1 register was not rewritten.
Note 2. Equivalent writing

Figure 23.49 Example of Basic Operation Timing in PWM Mode (2/2)

NOTES

1. D00, D01, D02, and D03: Set point of TSG2nCMP0 (0000_H - $FFFF_H$)
   D10, D11, D12, and D13: Set point of TSG2nCMP1 (0000_H - $FFFF_H$)
   D20, D21, D22, and D23: Set point of TSG2nCMP2 (0000_H - $FFFF_H$)
2. Outputs from the TSOn2 to TSOn6 pins behave similarly to the TSOn1 pin.
3. Δ: Write access

23.11.1.2 Interrupt/Reload Skipping Function in PWM Mode

By setting TSG2nCTL4.TSG2nPRE and TSG2nPIE to 1 and setting TSG2nRCC04 to TSG2nRCC00 and TSG2nCTL3.TSG2nRIA, the reload and interrupt skipping function can be used.

By setting TSG2nPRE to 1 and setting TSG2nRCC04 to TSG2nRCC00, the interrupt skipping function can be used.

23.11.1.3 Controlling Dead Time in PWM Mode

By setting the dead time in the TSG2nDTC0 and TSG2nDTC1 registers in PWM mode, it is possible to control the dead time. The dead time is controlled according to the switch timing of the TSOn1 and TSOn2 pin outputs, the TSOn3 and TSOn4 pin outputs, and the TSOn5 and TSOn6 pin outputs.

Table 23.51 Dead Time in PWM Mode

Switch Timing	Dead Time
TSOn1 : High level to low level TSOn2 : Low level to high level	Value of TSG2nDTC1 register
TSOn2 : High level to low level TSOn1 : Low level to high level	Value of TSG2nDTC0 register
TSOn3 : High level to low level TSOn4 : Low level to high level	Value of TSG2nDTC1 register
TSOn4 : High level to low level TSOn3 : Low level to high level	Value of TSG2nDTC0 register
TSOn5 : High level to low level TSOn6 : Low level to high level	Value of TSG2nDTC1 register
TSOn6 : High level to low level TSOn5 : Low level to high level	Value of TSG2nDTC0 register

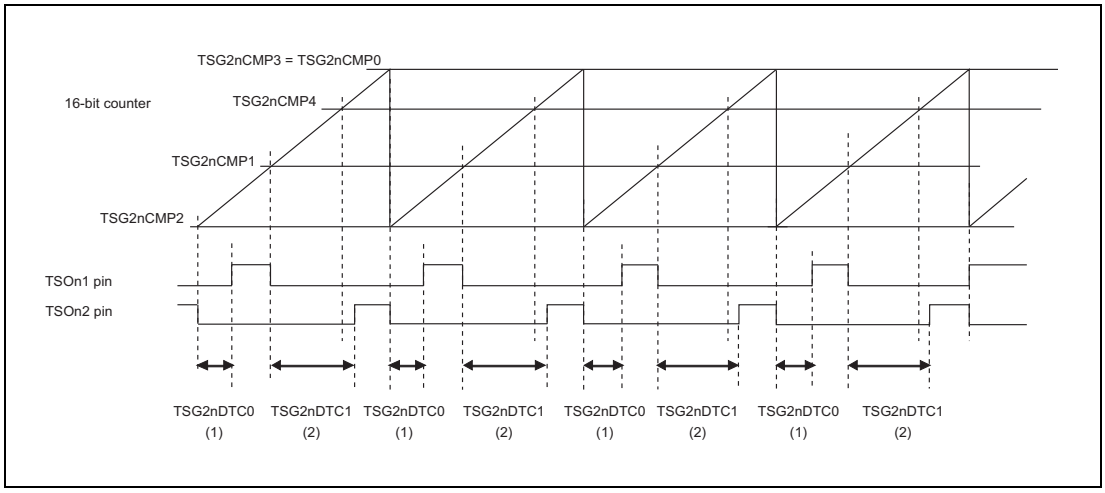


Figure 23.50 Example of Dead Time Control between TSOn1 and TSOn2 Outputs (1/2)

At (1), the dead time counter starts counting at the falling edge of the TSOn2 output. Here, although the 16-bit counter is 0000_H, the TSOn1 output stays inactive because the dead time counter is still operating. The TSOn1 output becomes active at the timing when the dead time count operation ends.

At (2), the dead time counter starts counting at the falling edge of the TSOn1 output. Even after the match of the 16-bit counter and TSG2nCMP4, the TSOn2 output stays inactive because the dead time counter is still operating. The TSOn2 output becomes active at the timing when the dead time count operation ends.

NOTES

1. The TSOn1 and TSOn2 pin outputs are active high.
2. The TSOn3 to TSOn6 pin outputs behave similarly.

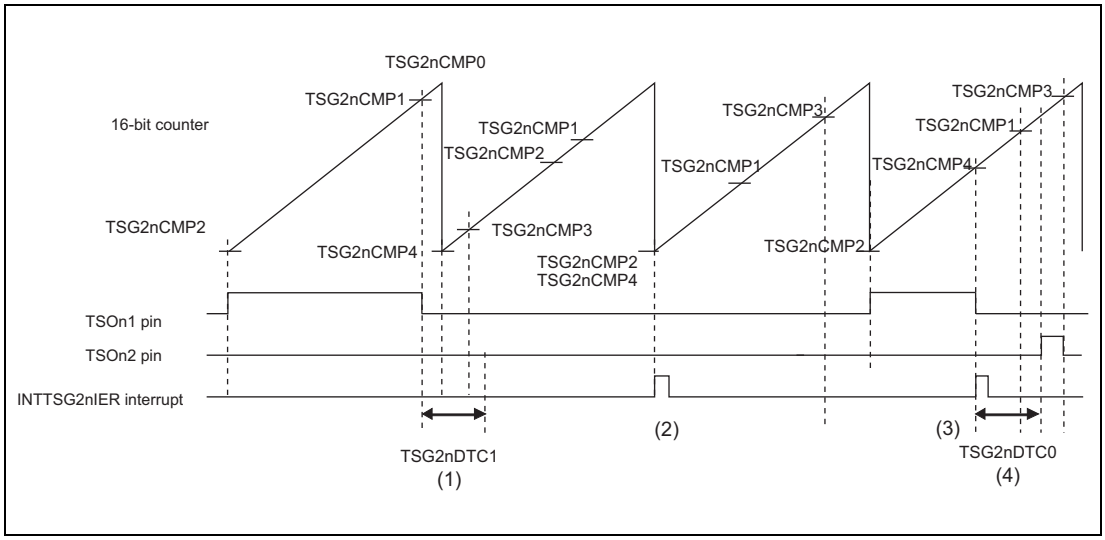


Figure 23.51 Example of Dead Time Control between TSOn1 and TSOn2 Outputs (2/2)

During (1), the dead time counter starts counting at the falling edge of the TSOn1 output. Even after the 16-bit counter reaches 0000_H and the match occurs between the 16-bit counter and TSG2nCMP4, the TSOn2 output stays inactive because the dead time counter is still operating. Moreover, since the TSG2nCMP3 register compare match occurs before the operation of the dead time counter ends, the TSOn2 output stays inactive.

$$TSG2nCMP1 + TSG2nDTC1 \geq TSG2nCMP0 + TSG2nCMP2 \text{ (TSOn2 stays inactive)}$$

$$TSG2nCMP2 + TSG2nDTC0 \geq TSG2nCMP0 + TSG2nCMP1 \text{ (TSOn1 stays inactive)}$$

At (2), the INTTSG2nIER interrupt occurs because the TSG2nCMP2 register and the TSG2nCMP4 register are set so that the TSOn1 and TSOn2 outputs rise simultaneously. Here, the TSOn1 output and the TSOn2 output are inactive.

At (3), compare match with the TSG2nCMP4 register generates an INTTSG2nIER interrupt and both TSOn1 and TSOn2 outputs become inactive.

At (4), the falling edge (inactive) of the TSOn1 output is caused by the simultaneous active state detected and the dead time counter starts counting. After the end of the dead time counter operation, the TSOn2 output becomes active.

NOTES

1. TSOn1 and TSOn2 are set to active high.
2. The TSOn3 to TSOn6 pin outputs behave similarly.

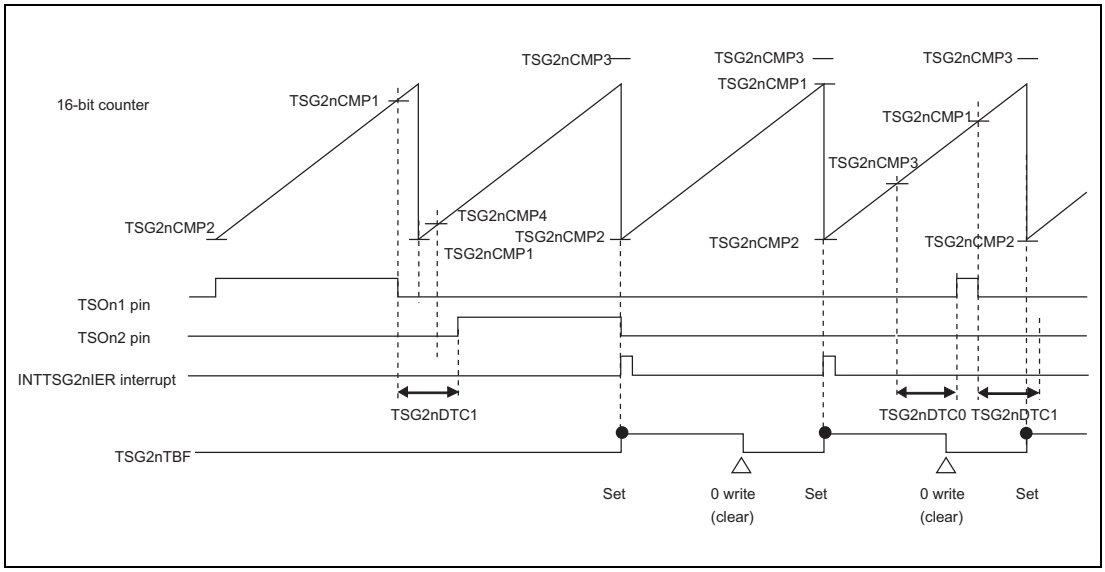


Figure 23.52 Example of 100% Duty Output at Dead Time Control

When the TSOn2 pin is set to duty cycle of 100% (TSG2nCMP3 ≥TSG2nCMP0 + 1), the output of the TSOn1 pin is fixed to a low level. This control is intended to mask the active condition of TSOn1 output since the TSOn2 output is active before the TSOn1 output becomes active. In this case, the INTTSG2nIER interrupt is also generated because TSOn1 and TSOn2 outputs become high simultaneously.

NOTES

1. TSOn1 and TSOn2 are set to active high.
2. The TSOn3 to TSOn6 pin outputs behave similarly.

23.11.1.4 Dead Time Rewriting during Timer Operation in PWM Mode

In PWM mode, it is possible to rewrite TSG2n dead time setting registers TSG2nDTC0 and TSG2nDTC1 while counting. The new settings are active at reload timing. It is not possible to change the dead time setting by rewriting at anytime.

Please enable reloading by writing to the TSG2nCMP1 register.

23.11.2 HT-PWM Mode (High Accuracy Triangular - Pulse Width Modulation Mode)

Overview

In this mode, the 16-bit counter (up/down count by ±2 bits, practically 15 bits) and the 16-bit compare registers (LSB is used to control additional pulse) are used to generate a 6-phase PWM signal.

Prerequisites

- Set the carrier wave period to TSG2nCMP0.
- Set the duty cycle of the voltage data signals of the U phase, V phase, and W phase with TSG2nCMPU, TSG2nCMPV, and TSG2nCMPW (The values set to TSG2nCMPU, TSG2nCMPV, and TSG2nCMPW are reflected immediately to the corresponding TSG2nCMPm (m = 1, 2, 5, 6, 9, 10)).
- Symmetric triangular wave control is described in this section. Please refer to **Section 23.11.2.10, Asymmetric Triangular Wave Control in HT-PWM Mode**, for asymmetric triangular wave control.

Functional description

Set the period of carrier wave and the duty cycle of the U phase, the V phase, and the W phase. Counting up begins when TSG2nTRG0.TSG2nTS is set to 1.

The 16-bit counter counts up from TSG2nDTC0 as the minimum value, and counts down upon the match of the maximum value of TSG2nCMP0 + TSG2nDTC0.

The dead time is set with TSG2nDTC0 and TSG2nDTC1. TSG2nDTC0 sets the dead time of inverse phase (OFF) to positive phase (ON) switch and TSG2nDTC1 sets the dead time of positive phase (OFF) to inverse phase (ON) switch. The 10-bit counters (TSG2nDTT1 to TSG2nDTT3) for dead time generation load the set values of TSG2nDTC0 and TSG2nDTC1 by the compare match of the 16-bit counter and the TSG2nCMPm buffer register (m = 1, 2, 5, 6, 9, 10), and start down-counting.

The INTTSG2nIm interrupts (m = 1, 2, 5, 6, 9, 10) are generated by the compare match of the 16-bit counter with the TSG2nCMP1, TSG2nCMP2, TSG2nCMP5, TSG2nCMP6, TSG2nCMP9, and TSG2nCMP10 buffer registers.

NOTE

The HT-PWM mode is enabled when TSG2nTRG0.TSG2nMD1 and TSG2nMD0 = 01_B.

23.11.2.1 Block Diagram and Basic Timing Chart

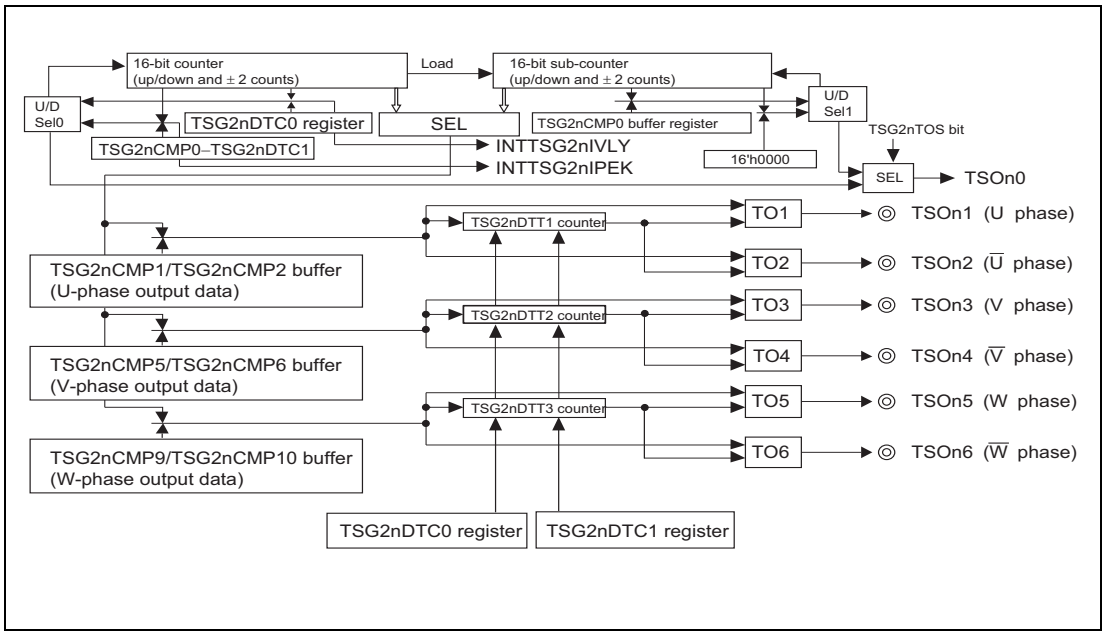


Figure 23.53 Block Diagram in HT-PWM Mode

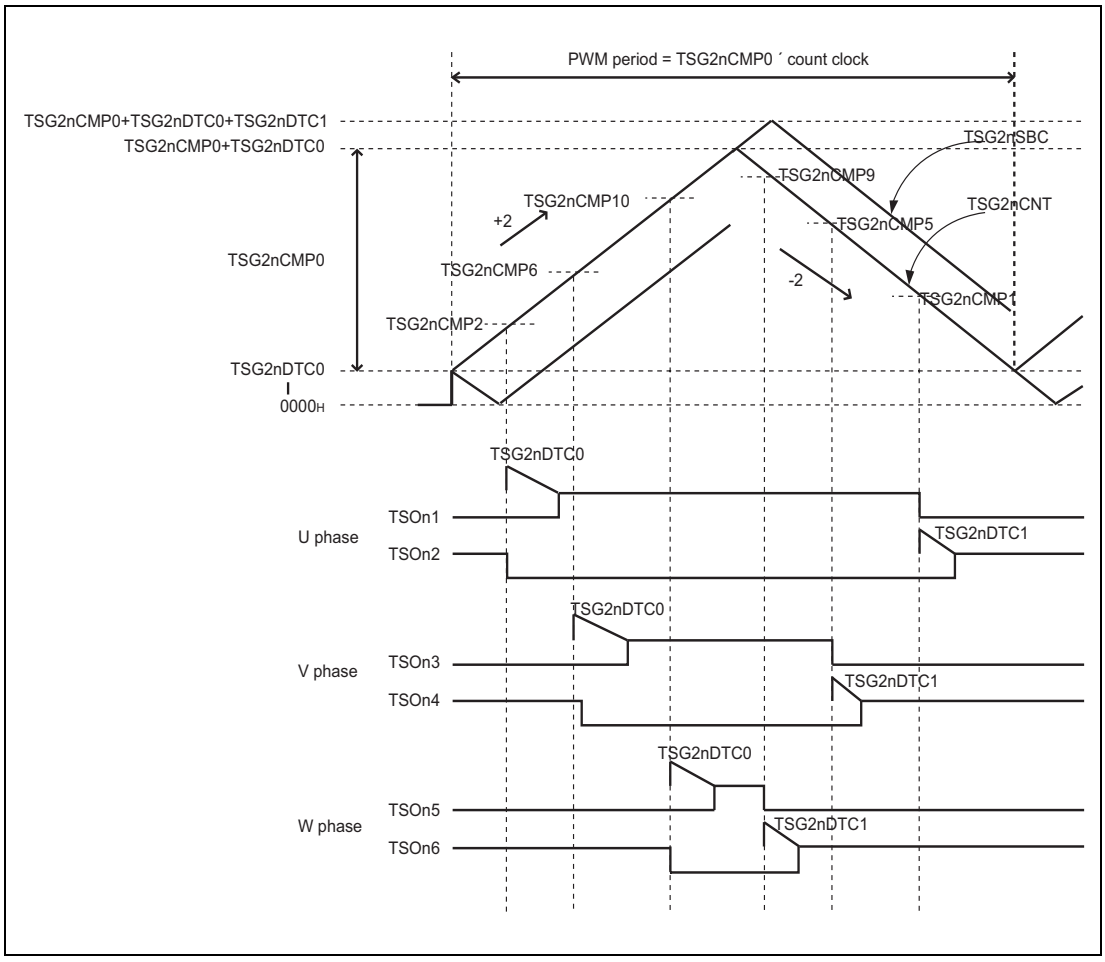


Figure 23.54 Basic Timing in HT-PWM Mode

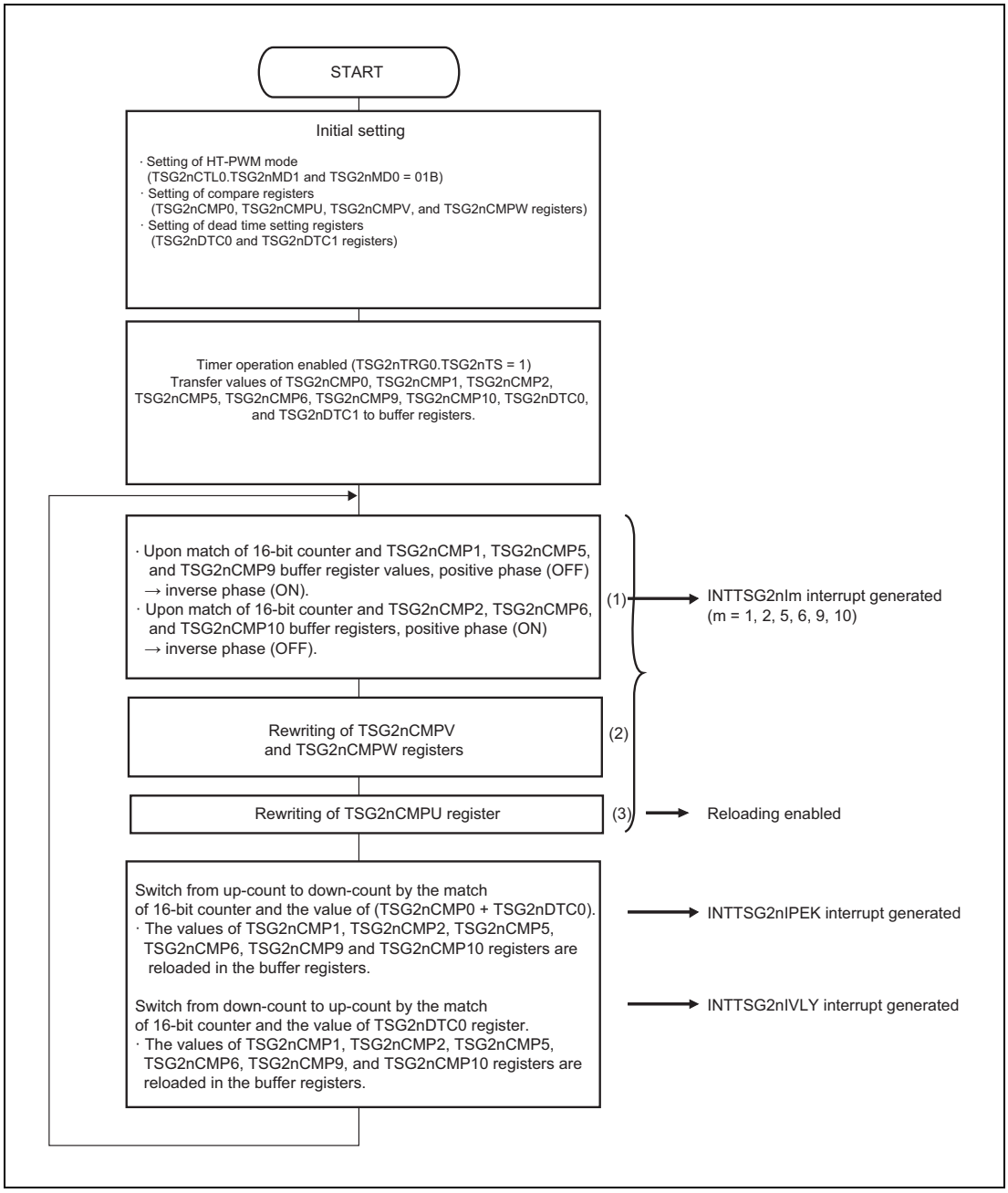


Figure 23.55 Basic Operation Flow in HT-PWM Mode

NOTES

- Write access to TSG2nCMPU (TSG2nCMP1) includes reloading enabling operation. Therefore, (3) must be done after (2).
- The INTTSG2nIPEK interrupt is generated only when TSG2nCTL4.TSG2nPRE = 1.
- The INTTSG2nIVLY interrupt is generated only when TSG2nCTL4.TSG2nVRE = 1.

23.11.2.2 List of HT-PWM Mode Operations

Table 23.52 Counter Function in HT-PWM Mode

Operation		Setting Condition
16-bit counter	Start	TSG2nTRG0.TSG2nTS = 0 → 1 (up count from TSG2nDTC0)
	Up count	Compare match of TSG2nDTC0 buffer register and 16-bit counter
	Down count	Compare match of TSG2nCMP0 + TSG2nDTC0 buffer register and 16-bit counter
	Clear	—
	Stop	TSG2nTRG1.TSG2nTT = 0 → 1
16-bit sub-counter	Start	TSG2nTRG0.TSG2nTS = 0 → 1 (down count from TSG2nDTC0)
	Up count	Underflow
	Down count	Compare match of TSG2nCMP0 + TSG2nDTC0 + TSG2nDTC1 buffer register and 16-bit sub-counter
	Load	• TSG2nCMP0 + TSG2nDTC0 when the value of 16-bit counter matches the value of buffer register TSG2nCMP0 + TSG2nDTC0 • TSG2nDTC0 when the value of 16-bit counter matches the value of the buffer register TSG2nDTC0
	Clear	—
	Stop	TSG2nTRG1.TSG2nTT = 0 → 1

Table 23.53 Compare Register and Dead Time Setting Register Functions in HT-PWM Mode

Register	Rewrite Method	Rewrite during Operation	Function
TSG2nCMP0	Reload/Anytime rewrite	Possible	Setting period
TSG2nCMPU	—	Possible	PWM control for U phase
TSG2nCMP1W (TSG2nCMP1, TSG2nCMP2)	Reload/Anytime rewrite		
TSG2nCMPV	—	Possible	PWM control for V phase
TSG2nCMP5W (TSG2nCMP5, TSG2nCMP6)	Reload/Anytime rewrite		
TSG2nCMPW	—	Possible	PWM control for W phase
TSG2nCMP9W (TSG2nCMP9, TSG2nCMP10)	Reload/Anytime rewrite		
TSG2nDCMP0W, TSG2nDCMP2	Reload/Anytime rewrite	Possible	Diagnostic signal output or A/D conversion trigger
TSG2nDTC0, TSG2nDTC1	Reload	Possible conditionally	Period and dead time setting

NOTES

- The values written to TSG2nCMPU, TSG2nCMPV, and TSG2nCMPW registers are set to both the lower and upper 16-bits of the respective TSG2nCMP1W (TSG2nCMP1, TSG2nCMP2), TSG2nCMP5W (TSG2nCMP5, TSG2nCMP6), and TSG2nCMP9W (TSG2nCMP9, TSG2nCMP10) registers.
- Please refer to **(1), TSG2nDTC0 and TSG2nDTC1 Rewriting**, on how to rewrite the TSG2nDTC0 and TSG2nDTC1.

Table 23.54 Timer Output Function in HT-PWM Mode

Pin	Function
TSOn0	Active level output during up count, inactive level output at down count of the 16-bit counter/sub-counter
TSOn1	PWM output with dead time by compare match of TSG2nCMP1 buffer register and 16-bit counter (down count) and compare match of TSG2nCMP2 buffer register and 16-bit counter (up count)
TSOn2	Inverse phase output to TSOn1
TSOn3	PWM output with dead time by compare match of TSG2nCMP5 buffer register and 16-bit counter (down count) and compare match of TSG2nCMP6 buffer register and 16-bit counter (up count).
TSOn4	Inverse phase output to TSOn3
TSOn5	PWM output with dead time by compare match of TSG2nCMP9 buffer register and 16-bit counter (down count) and TSG2nCMP10 buffer register and 16-bit counter (up count).
TSOn6	Inverse phase output to TSOn5
TSOn7	Diagnostic signal output or pulse output by A/D conversion trigger

NOTE

State of TSOn0 output can be switched with TSG2nIOC1.TSG2nTOS.

Table 23.55 Interrupt Request in HT-PWM Mode

Interrupt	Function
INTTSG2nI0	Compare match of TSG2nDTC0 buffer register and 16-bit counter (periodic interrupt)
INTTSG2nIm (m = 1, 2, 5, 6, 9, 10)	Compare match of TSG2nCMPm buffer register and 16-bit counter
INTTSG2nIER	Error interrupt
INTTSG2nIVLY	Trough interrupt
INTTSG2nIPEK	Peak interrupt
INTTSG2nIWN	Warning interrupt

Table 23.56 Compare Match Timing in HT-PWM Mode

Compare Match	Timing
TSG2nCMP0	When 16-bit counter changes from TSG2nCMP0 + TSG2nDTC0 to TSG2nCMP0 + TSG2nDTC0 - 2
TSG2nCMPm (m = 1, 2, 5, 6, 9, 10)	When 16-bit counter changes from TSG2nCMPm to TSG2nCMPm ± 2 (m = 1, 2, 5, 6, 9, 10)

Table 23.57 Example of Setting Each Timer Output Condition in HT-PWM Mode

Pin	Item	Output Period	Output Duty	
			Output Condition	Setting Condition
TSOn0	Toggle output	TSG2nCMP0 × count clock	Output an active level during up count, and an inactive level during down count	—
TSOn1, TSOn3, TSOn5	PWM output	TSG2nCMP0 × count clock	Output an inactive level throughout one period (0% duty)	TSG2nCMPm = TSG2nCMP0 + TSG2nDTC0 + TSG2nDTC1 (m = U, V, W)
			Output an active level of one count clock in one period	TSG2nCMPm = TSG2nCMP0 - 1 (m = U, V, W)
			Output an inactive level of one count clock in one period	TSG2nCMPm = 0001_H (m = U, V, W)
			Active level during all period (100% duty)	TSG2nCMPm = 0000_H (m = U, V, W)
TSOn2, TSOn4, TSOn6	PWM output		Inactive level during all period (0% duty)	TSG2nCMPm = 0000_H (m = U, V, W)
			Output an active level of one count clock in one period	TSG2nCMPm = TSG2nDTC0 + TSG2nDTC1 + 1 (m = U, V, W)
			Output an inactive level of one count clock in one period	TSG2nCMPm = TSG2nCMP0 + TSG2nDTC0 + TSG2nDTC1 - 1 (m = U, V, W)
			Active level during all period (100% duty)	TSG2nCMPm = TSG2nCMP0 + TSG2nDTC0 + TSG2nDTC1 (m = U, V, W)
TSOn7	Diagnostic signal output or pulse output by A/D conversion trigger	TSG2nCMP0 × count clock	Please refer to **Section 23.9.1, A/D Operation of A/D Conversion Trigger**.	

23.11.2.3 Various Settings of HT-PWM Mode

Mode Setting

HT-PWM can be used when TSG2nCTL0.TSG2nMD1 and TSG2nMD0 are set to 01_B.

Setting timer output

The output pins TSOn1 to TSOn6 are controlled by setting TSG2nIOC0, TSG2nIOC2, and TSG2nIOC3.

The output pin TSOn0 indicates the up/down count status of the 16-bit counter or the 16-bit sub-counter. Switch between the 16-bit sub-counter and the 16-bit counter is done with the TSG2nIOC1.TSG2nTOS bit.

The TSOn7 pin outputs pulses of the diagnostic output or the A/D conversion trigger. Please set it as required.

Enabling error interrupt generation

Error interrupt (INTTSG2nIER) due to the detection of the simultaneous active state of the positive phase and inverse phase is enabled by setting TSG2nIOC1.TSG2nEOC to 1. In HT-PWM mode, with any value set in the compare register, the simultaneous active state of the positive phase and inverse phase is not possible. Please refer to **Section 23.10, Error and Warning Interrupts**, for details.

Setting register rewriting timing with reload function

With TSG2nCTL3.TSG2nRMC, reload (simultaneous rewrite) or rewrite (anytime) is specified for the registers with reload function. The default setting is 0 (reload). To reload, set TSG2nCTL4.TSG2nPRE or TSG2nVRE to 1.

The reload timing is not generated if both the TSG2nPRE and TSG2nVRE bits are set to 0.

When "anytime rewrite" is specified, the unintended output may be generated depending on the rewrite timing.

Setting interrupts and skipping function

Interrupts and the skipping function are set with TSG2nCTL4. TSG2nPIE should be set to 1 when peak interrupt (INTTSG2nIPEK) is necessary and TSG2nVIE to 1 when trough interrupt (INTTSG2nIVLY) is necessary. To use the skipping function for peak/trough interrupts, the TSG2nRCC04 to TSG2nRCC00 must be set.

Setting A/D conversion trigger output

To set A/D conversion trigger 0 (TSTADT0 signal), use TSG2nCTL5.TSG2nAT09 to TSG2nAT00.

With TSG2nAT09 to TSG2nAT00, A/D conversion trigger output is enabled or disabled at the match of 16-bit counter (during up count) with TSG2nDCMP2 to TSG2nDCMP0, the match of the 16-bit counter (during down count) with TSG2nDCMP2 to TSG2nDCMP0, the 16-bit counter peak interrupt (INTTSG2nIPEK), the 16-bit counter trough interrupt (INTTSG2nIVLY), the 16-bit sub-counter peak timing, and 16-bit sub-counter trough timing.

To set A/D conversion trigger 1 (TSTADT1 signal), use TSG2nCTL6.TSG2nAT19 to TSG2nAT10.

To set the match timing of 16-bit counter and TSG2nDCMP2 to TSG2nDCMP0, set the compare value to the pertinent register.

The skipping function can be used for TSTADT0 and the TSTADT1 signals. Use TSG2nACC00, TSG2nACC01 of TSG2nCTL5, TSG2nACC10, and TSG2nACC11 of TSG2nCTL6 to select the skipping rate among 1/1, 1/2, 1/4, and 1/8.

CAUTION

Set TSG2nCTL5, TSG2nCTL6, and TSG2nDCMP2 to TSG2nDCMP0 correctly when using the TSOn7 output for the A/D conversion trigger timing pulse.

Setting dead time

The dead time can be set with TSG2nDTC0 and TSG2nDTC1.

The dead time is calculated by the following expressions:

PCLK × TSG2nDTC0

PCLK × TSG2nDTC1

TSG2nDTC0 can set the time between a change of TSOn2, TSOn4, and TSOn6 to the inactive state and a change of TSOn1, TSOn3, and TSOn5 to the active state, respectively.

TSG2nDTC1 can set the time between a change of TSOn1, TSOn3, TSOn5 to the inactive state and a change of TSOn2, TSOn4, and TSOn6 to the active state, respectively.

TSG2nDTC0 and TSG2nDTC1 can only be set to an even value.

Carrier period

Set the carrier period with TSG2nCMP0 according to the following expression:

TSG2nCMP0 = Carrier period/count clock period (PCLK)

Satisfy the following requirements when setting the TSG2nCMP0 register regarding the dead time:

- TSG2nCMP0 + TSG2nDTC0 + TSG2nDTC1 ≤ FFFEH
- TSG2nCMP0 > TSG2nDTC0
- TSG2nCMP0 > TSG2nDTC1
- TSG2nCMP0 > 3 × MAX (TSG2nDTC0, TSG2nDTC1)
- TSG2nCMP0: Even number

NOTES

MAX (A, B) indicates the larger value of A and B.

Duty (PWM width) setting

The duty of the U phase, the V phase, and the W phase is set with TSG2nCMPm (m = U, V, W, or 1, 2, 5, 6, 9, and 10), respectively. The setting range of the compare registers is as follows:

0000_H ≤ TSG2nCMPm ≤ TSG2nCMP0 + TSG2nDTC0 + TSG2nDTC1

LSB (least significant bit) of TSG2nCMPU, TSG2nCMPV, and TSG2nCMPW indicates the setting of an additional pulse. When TSG2nCMPU = 0003_H, the change in the inverse phase (TSOn2 output) is done one count clock later compared to the TSG2nCMPU = 0002_H setting (when the 16-bit counter is up-counting). The additional pulse cannot be set to TSG2nCMP1, TSG2nCMP2, TSG2nCMP5, TSG2nCMP6, TSG2nCMP9, or TSG2nCMP10 (only even numbers can be set to these registers).

23.11.2.4 16-Bit Counter Operation in HT-PWM Mode

The 16-bit counter is initialized to 0000_H and the value of TSG2nDTC0 is loaded immediately after starting the TSG2n timer operation (TSG2nTRG0.TSG2nTS = 1). Afterwards, counting is done by +2. After 16-bit counter reaches the value of TSG2nCMP0 + TSG2nDTC0, counting is done by -2.

Figure 23.56 shows 16-bit counter operation.

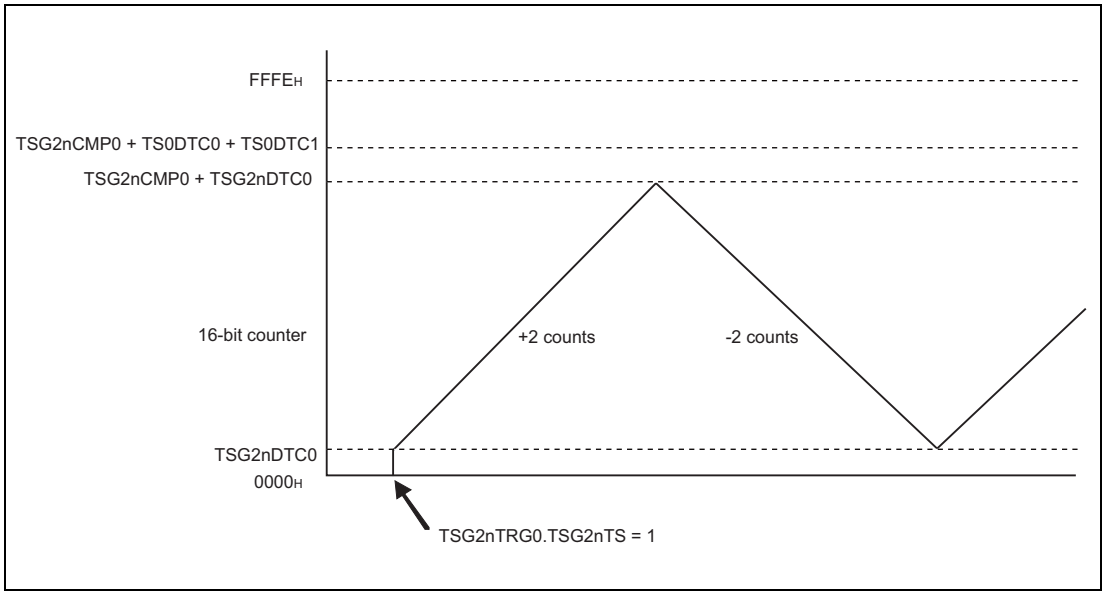


Figure 23.56 Example of 16-Bit Counter Operation in HT-PWM Mode

NOTE

Minimum 16-bit counter value: TSG2nDTC0
Maximum 16-bit counter value: TSG2nCMP0 + TSG2nDTC0
Carrier period : TSG2nCMP0 × count clock period (PCLK)

The 16-bit sub-counter is initialized to 0000_H and the value of TSG2nDTC0 is loaded immediately after starting the TSG2n timer operation (TSG2nTRG0.TSG2nTS = 1). Afterwards, counting by -2 is done until 0000_H is reached and counting by +2 begins. Next, the value of the 16-bit counter is loaded into the 16-bit sub-counter at a change timing of the 16-bit counter into the down count. Counting up by the 16-bit sub-counter continues until the value reaches the value of TSG2nCMP0 + TSG2nDTC0 + TSG2nDTC1, and then counting by -2 begins. Similarly, when the 16-bit counter value matches the TSG2nDTC0 value, the 16-bit counter value is loaded to the 16-bit sub-counter and the down count is continued.

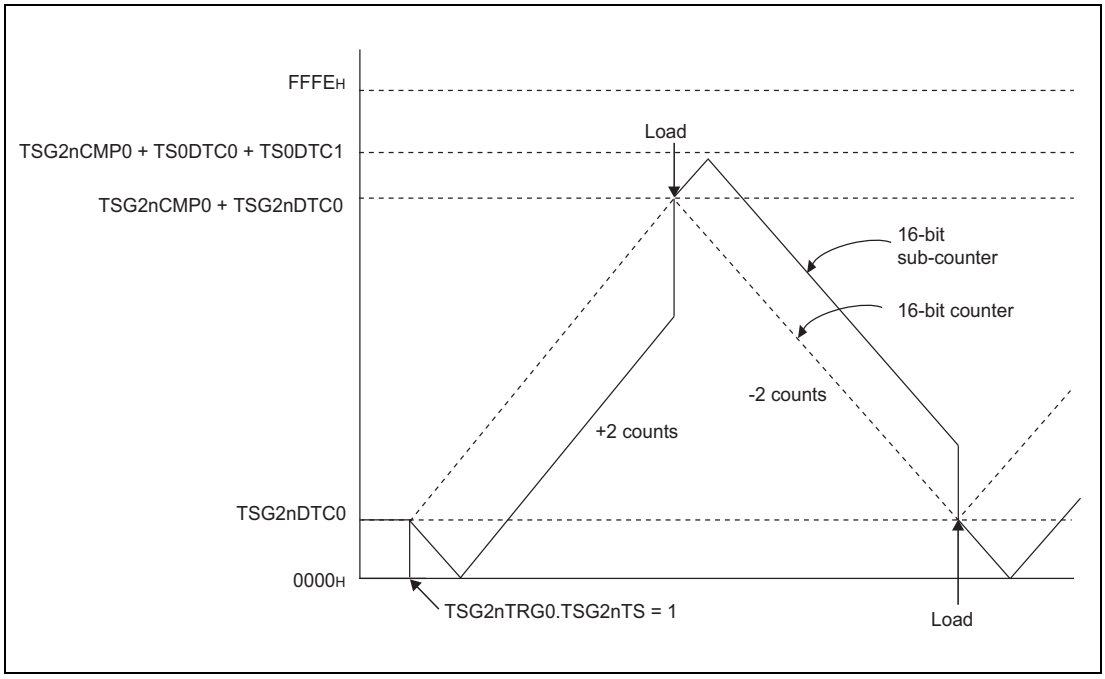


Figure 23.57 Example of 16-bit Sub-Counter Operation in HT-PWM Mode

NOTE

Minimum 16-bit sub-counter value: 0000_H
Maximum 16-bit sub-counter value: TSG2nCMP0 + TSG2nDTC0 + TSG2nDTC1

23.11.2.5 Basic Operation of HT-PWM Mode

(1) Example of Timer Output Immediately after the Start of the TSG2n Timer Operation

Figure 23.58 shows the timing chart when TSG2nCMP0 = $000E_H$, TSG2nDTC0 = 0002_H, TSG2nDTC1 = 0004_H and TSG2nCMPU = 0000_H-0014_H (excerpt). In this example, TSG2nIOC2.TSG2nOL1-TSG2nOL6 = 000000_B.

In the case of TSG2nCMPU = TSG2nDTC0, when the initial value of the counter is loaded after TSG2n begins operation (TSG2nTRG0.TSG2nTS = 1), the TSOn2 output is changed to the active level. When TSG2nCMPU = TSG2nDTC0-0001_H, an additional pulse is generated, the TSOn2 output is changed one count clock cycle later compared to the case of TSG2nCMPU = TSG2nDTC0-0002_H.

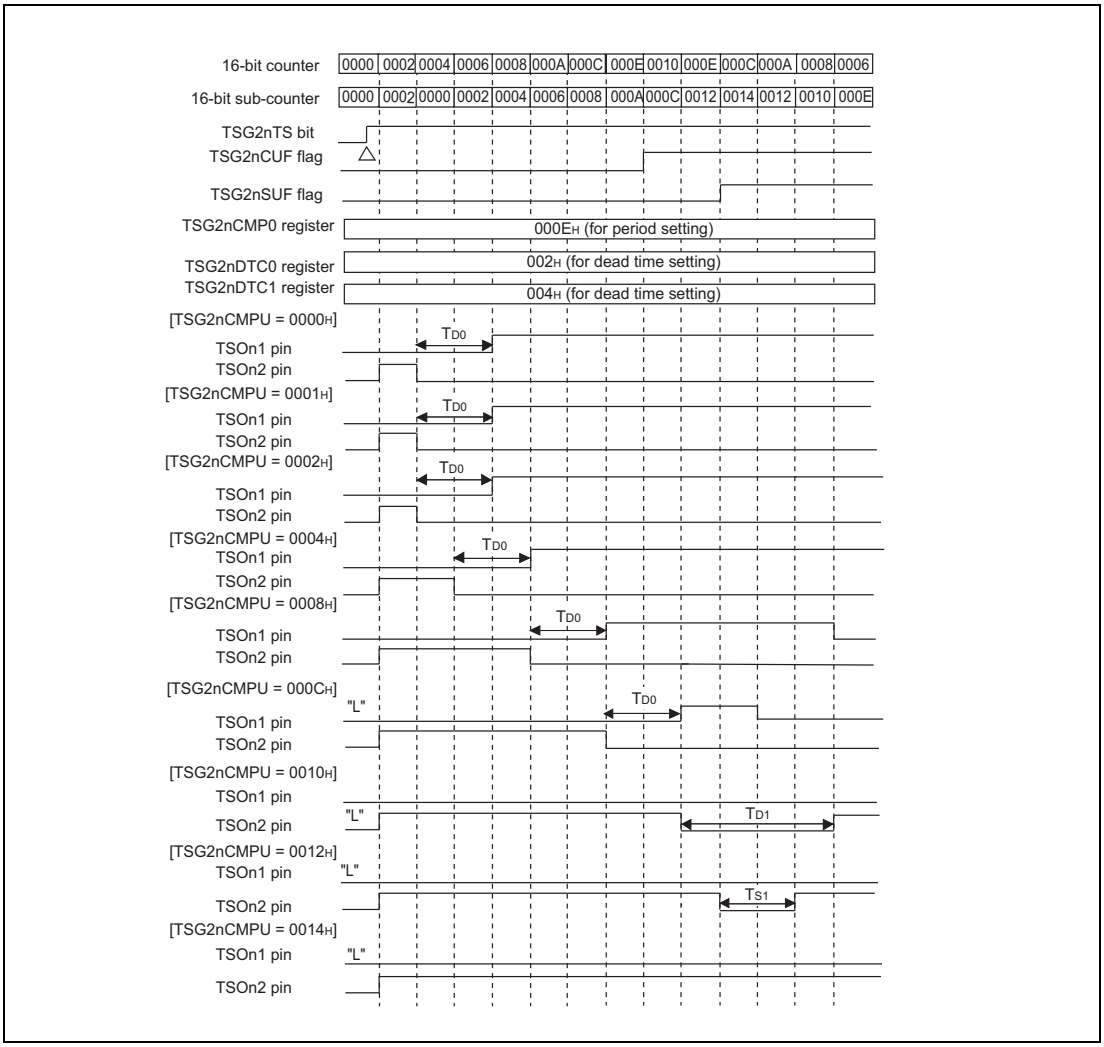


Figure 23.58 Example of Timer Output when TSG2nTS is Set to 1 (Initial) in HT-PWM Mode

NOTES

1. TSG2nCMP0 = $000E_H$, TSG2nDTC0 = 0002_H, TSG2nDTC1 = 0004_H
2. TD0: Time depending on setting of the dead time in the TSG2nDTC0 register
   TD1: Time depending on setting of the dead time in the TSG2nDTC1 register
   TS1: Time decided by compare match of the 16-bit sub-counter and TSG2nCMPU register, when TSG2nCMPU >16-bit counter maximum value
3. Δ: Write access

(2) Example of Timer Output during TSG2n Timer Operation

The figure below shows the timing chart when TSG2nCMP0 = $000E_H$, TSG2nDTC0 = 0002_H, TSG2nDTC1 = 0004_H, and TSG2nCMPU is set to 0000_H-0014_H (excerpt). In this example, TSG2nIOC2.TSG2nOL1-TSG2nOL6 = 000000_B.

The range of the active (high level) width of a positive phase (TSOn1) output is $0000_H \leq$ TSG2nCMPU $\leq$ TSG2nCMP0 (for the additional pulse). The range of the active (high level) width of an inverse phase (TSOn2) output is TSG2nDTC0 + TSG2nDTC1 $\leq$ TSG2nCMPU $\leq$ TSG2nCMP0 + TSG2nDTC0 + TSG2nDTC1.

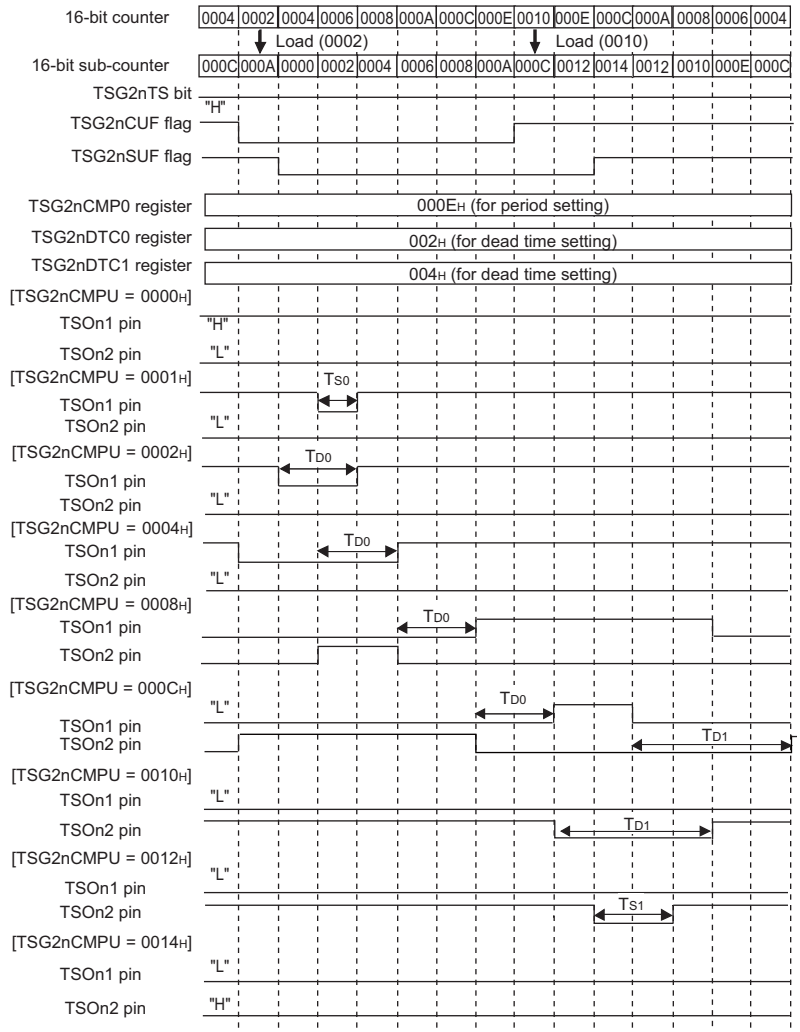


Figure 23.59 Example of Timer Output during TSG2n Operation in HT-PWM Mode

NOTES

1. TSG2nCMP0 = $000E_H$, TSG2nDTC0 = 0002_H, TSG2nDTC1 = 0004_H
2. TD0 : Time depending on setting of the dead time in the TSG2nDTC0 register
   TD1 : Time depending on setting of the dead time in the TSG2nDTC1 register
   TS0 : Time decided by compare match of 16-bit sub-counter and the TSG2nCMPU register, when TSG2nCMPU < 16-bit counter minimum value
   TS1 : Time decided by compare match of the 16-bit sub-counter and the TSG2nCMPU register, when TSG2nCMPU > 16-bit counter maximum value

23.11.2.6 Additional Pulse Control in HT-PWM Mode

The HT-PWM mode can generate an additional pulse by setting 1 to the LSB of the duty setting registers (TSG2nCMPU, TSG2nCMPV, and TSG2nCMPW). This allows more precise control of the pulse duty than standard pulse control.

The following sections describe two examples of pulse output of TSOn1: additional pulse control is used in one example and additional pulse control is not used in another.

(1) Example of Pulse Output when Additional Pulse Control is Used

Figure 23.60 shows the additional pulse control when an odd value is set to TSG2nCMPU.

The arrows and numerical values show the width of the duty cycle of the TSOn1 output in one period.

When the additional pulse control is used as shown in **Figure 23.60**, the width of the output of the TSOn1 (duty cycle) can be set within a range from the width of 12 clock cycles to the width of 0 clock cycles in one-clock-cycle step.

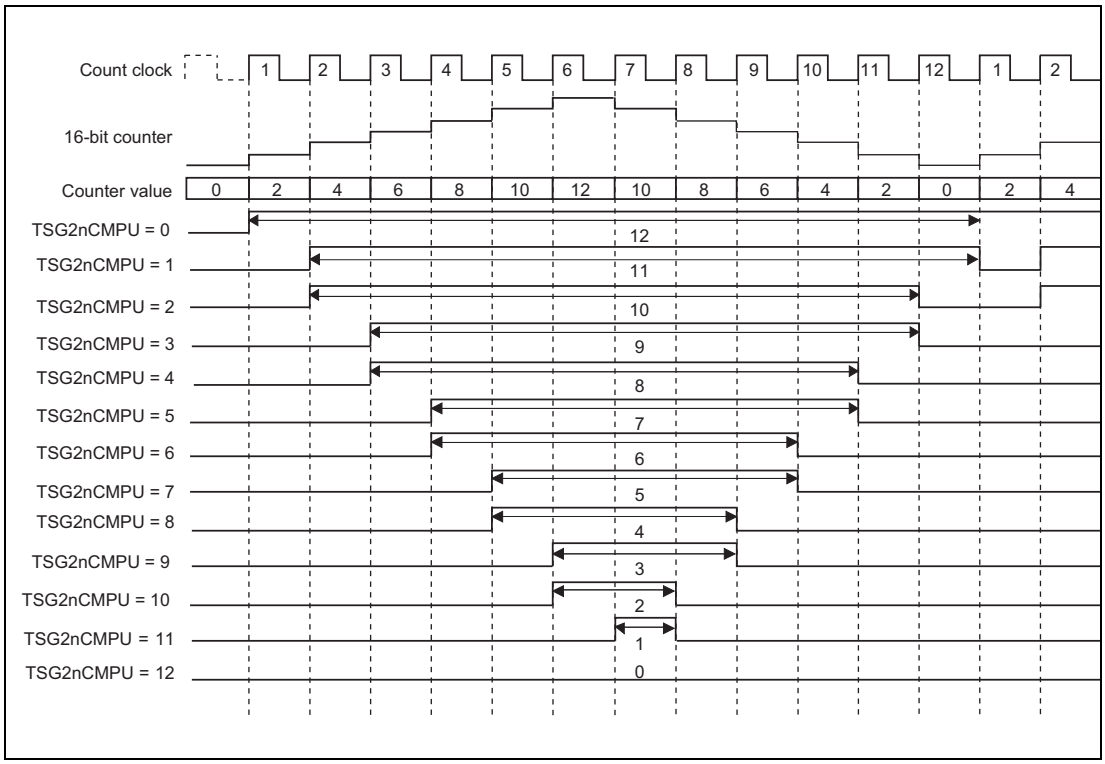


Figure 23.60 Example of TSOn1 Output when Additional Pulse Control is Used in HT-PWM Mode

NOTE

TSG2nCMP0 = 12, TSG2nDTC0 = 0, TSG2nDTC1 = 0

(2) Example of Pulse Output when Additional Pulse Control is Not Used

The arrows and numerical values in **Figure 23.61** show the width of the duty cycle of the TSOn1 output in one period.

When the additional pulse control is not used, the width of the TSOn1 output can be set within a range from the width of 12 clock cycles to the width of 0 clock cycles in two-clock-cycle step. In this case, the change in duty cycle is larger than that in the case when the additional pulse control is used.

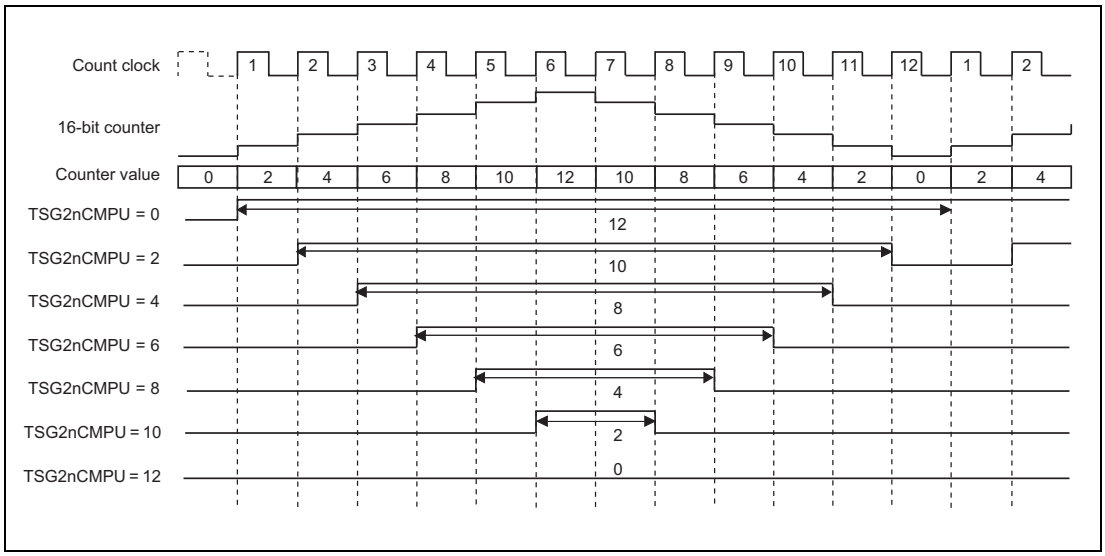


Figure 23.61 Example of Output when Additional Pulse Control is Not Used in HT-PWM Mode

NOTE

TSG2nCMP0 = 12, TSG2nDTC0 = 0, TSG2nDTC1 = 0

23.11.2.7 Dead Time Control in HT-PWM Mode

Duty setting registers are TSG2nCMP1, TSG2nCMP2, TSG2nCMP5, TSG2nCMP6, TSG2nCMP9, and TSG2nCMP10 and registers for setting the period are TSG2nCMP0, TSG2nDTC0, and TSG2nDTC1 in HT-PWM mode. The 6-phase PWM waveform of a variable duty is output by using these registers. To achieve the dead time control, there are two dead time setting registers (TSG2nDTC0 and TSG2nDTC1) and six 10-bit down counters that operate synchronously with the count clock of the 16-bit counter. TSG2nDTC0 is used for setting a dead time from a change of the inverse phase to the inactive state to a change of the positive phase to the active state. TSG2nDTC1 is used for setting a dead time from a change of the positive phase to the inactive state to a change of the inverse phase to the active state.

Figure 23.11 shows the output waveform when TSG2nDTC0 = x and TSG2nDTC1 = y.

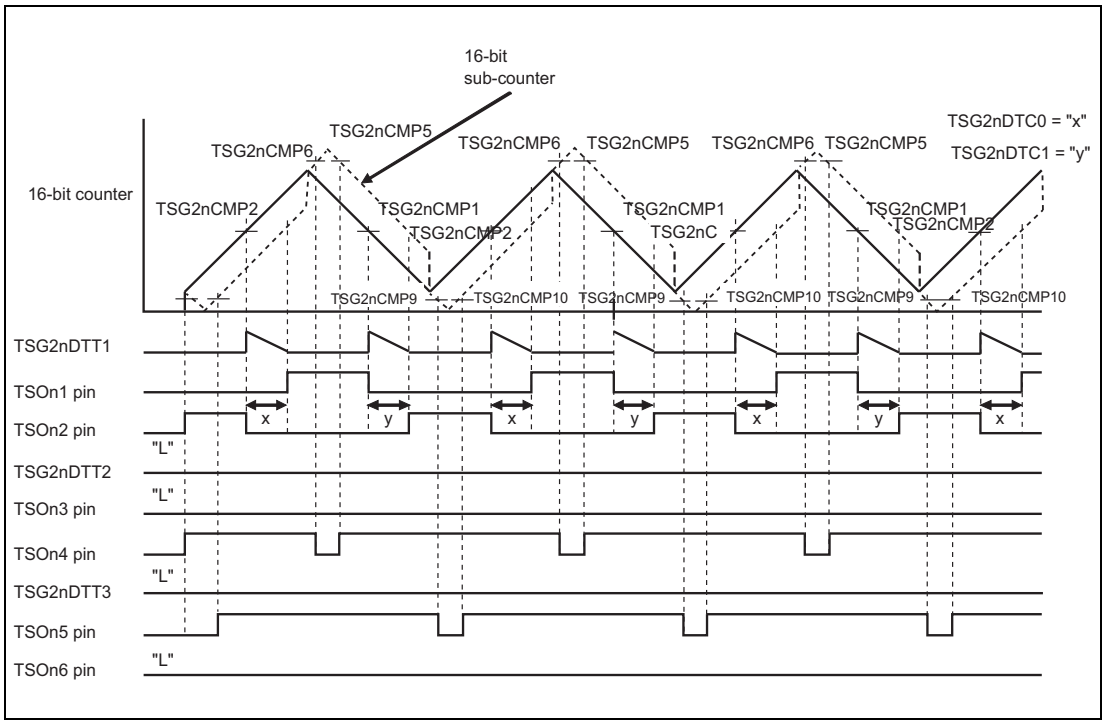


Figure 23.62 Example of Output Waveform with Dead Time in HT-PWM Mode

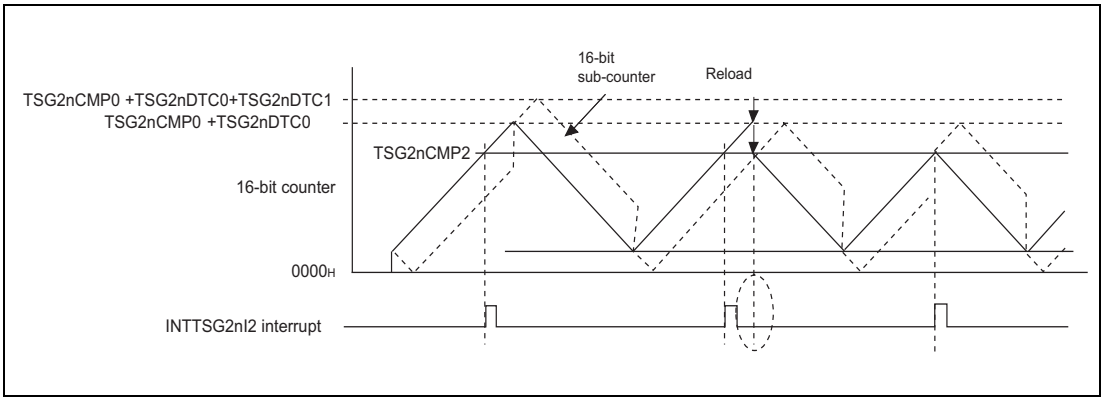
23.11.2.8 Notes Concerning Dead Time Control in HT-PWM Mode

(1) TSG2nDTC0 and TSG2nDTC1 Rewriting

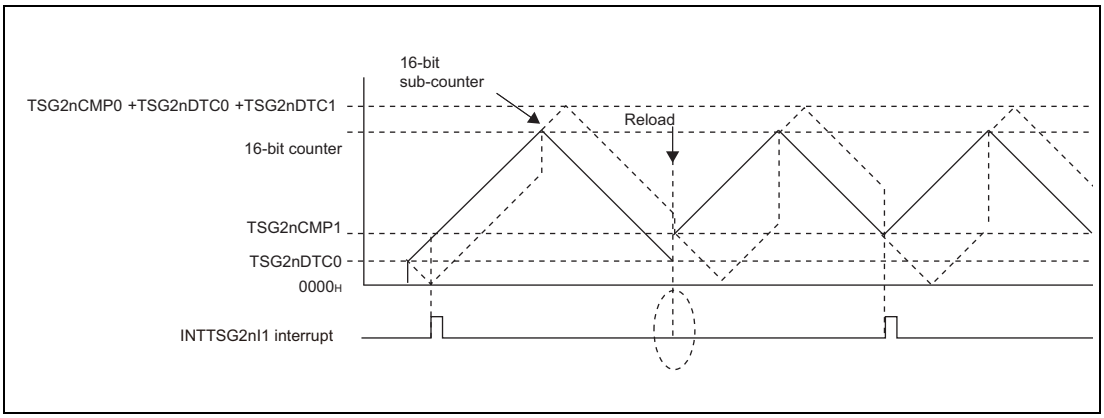
It is possible to rewrite the dead time setting in TSG2nDTC0 and TSG2nDTC1 registers during timer operation.

CAUTIONS

1. **Rewrite TSG2nDTC0 and TSG2nDTC1 when the reload function is used (TSG2nRMC = 0).**
2. **The write protection code check function is applied when TSG2nDTC0 and TSG2nDTC1 are rewritten. Refer to the pertinent register description for details.**
3. **When the TSG2nCMP0 and TSG2nDTC1 are updated at the peak of the 16-bit counter:**
   If the setting of TSG2nCMPm is greater than the updated TSG2nCMP0 + TSG2nDTC0 (new maximum value of the main counter), the match interrupt (INTTSG2nIm) will not be generated immediately after reload (m = 2, 6, 10).



4. **When the TSG2nDTC0 is updated at the trough of the 16-bit counter:**
   If the setting of TSG2nCMPm is less than the updated TSG2nDTC0 (new minimum value of the main counter), the match interrupt (INTTSG2nIm) is not generated immediately after reload (m = 1, 5, 9).



23.11.2.9 Software Output Control Function in HT-PWM Mode

TSG2nOPT0.TSG2nSOC, TSG2nIDC, and TSG2nOPT1.TSG2nSPC2-TSG2nSPC0 are used in HT-PWM mode for software control of timer output control.

As shown in **Figure 23.63**, with TSG2nSTE = 0, the output control is switched immediately when TSG2nSOC is set to 1. If the dead time is set, the period of the dead time is guaranteed. After that, when TSG2nSOC is set to 0, output control is retained. When the reload timing is generated, output control is switched to HT-PWM mode output control.

For details, refer to **Section 23.11.5, Software Output Control Function**.

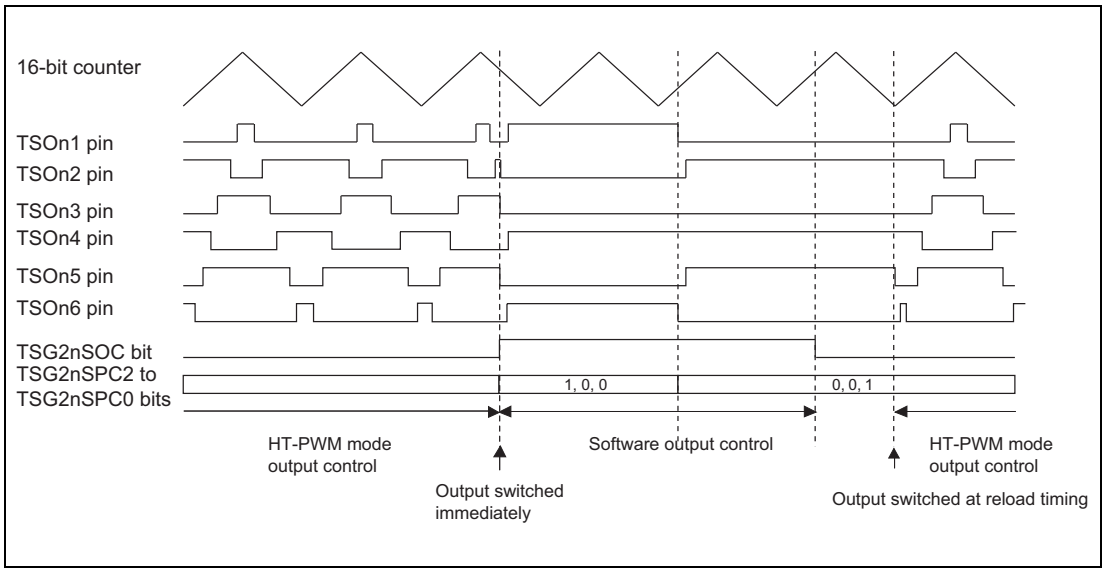


Figure 23.63 Example of Software Output Control Switching in HT-PWM Mode

CAUTION

Use reload (simultaneous rewrite) mode (TSG2nCTL3.TSG2nRMC = 0) when software output control function is used.

(1) Procedure for Software Output Control

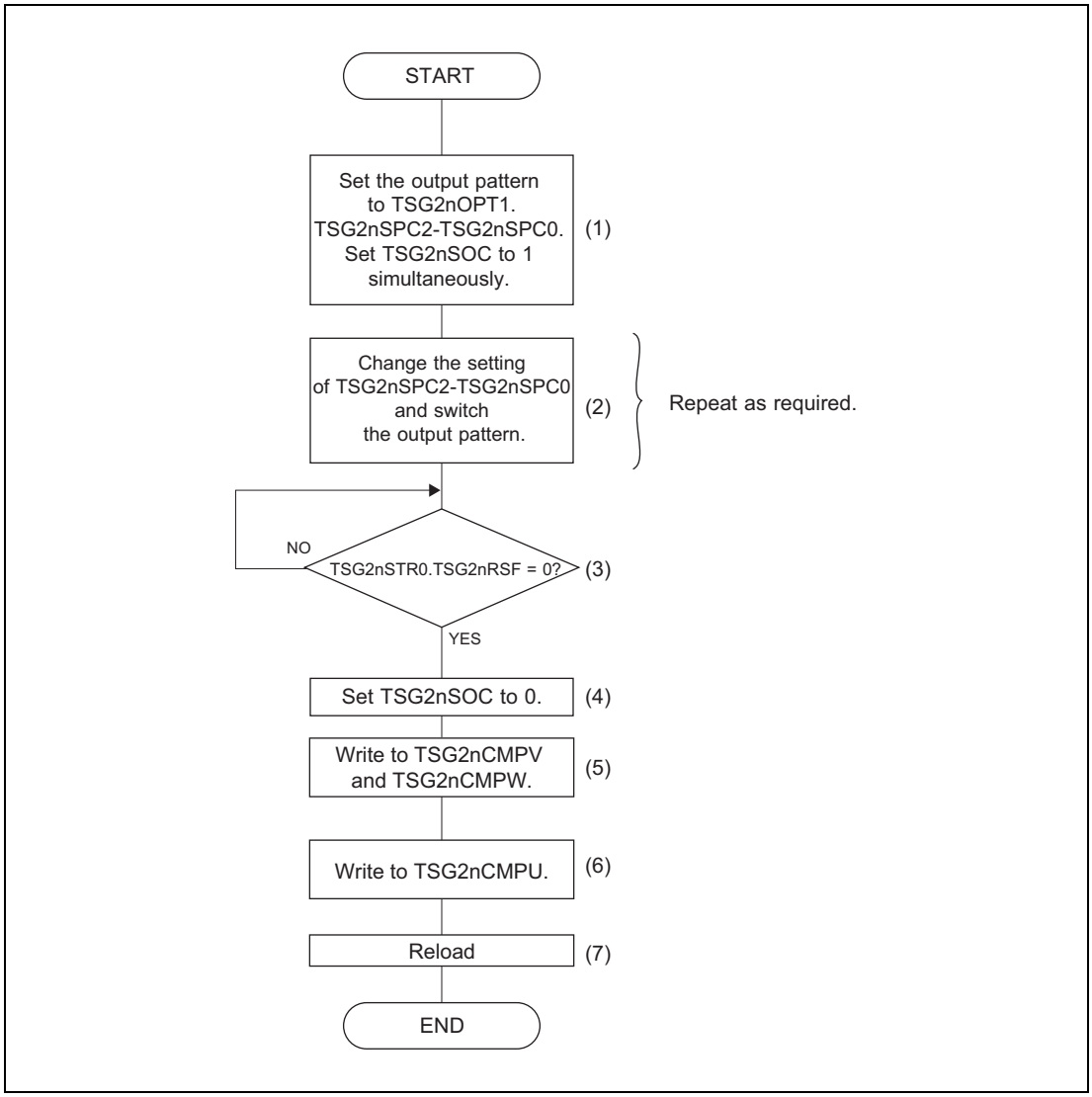


Figure 23.64 Flow of Software Output Control in HT-PWM Mode

The procedure for software output control is described below.

(1) Set the output pattern to the TSG2nOPT1.TSG2nSPC2-TSG2nSPC0. To enable software output control, set TSG2nOPT0.TSG2nSOC to 1 simultaneously.

(2) Change the output pattern setting of the TSG2nSPC2-TSG2nSPC0 to change the timer output. The registers that can be changed during software control are: TSG2nTRG1.TSG2nTT, TSG2nCTL3-TSG2nCTL6, TSG2nOPT0, TSG2nOPT1, TSG2nCMPm (m = 0, 1, 2, 5, 6, 9, 10), TSG2nDTC0, and TSG2nDTC1

(3) Confirm that reload request flag TSG2nSTR0.TSG2nRSF = 0. In case TSG2nRSF = 1, do not proceed to the following step until TSG2nRSF = 0.

(4) By setting TSG2nSOC = 0, the software control starts to be released (it is not released here yet).

(5) After releasing the software output control, set the necessary compare registers. Proceed to the following step when the register setting is not required. Here, change the registers with the reload function.

(6) Write to TSG2nCMPU (TSG2nCMP1) to start reloading.

(7) Reload is executed and software output is released.

CAUTION

Execute reload after executing steps (3), (4), (5), (6), and (7). When reload cannot be executed, the software output cannot be released.

23.11.2.10Asymmetric Triangular Wave Control in HT-PWM Mode

In HT-PWM mode, it is possible to control output by an asymmetric triangular waveform by setting the different timings for setting and clearing the U, V, and W phases.

The following describes the differences of the asymmetric triangular wave control from the symmetric triangular wave control.

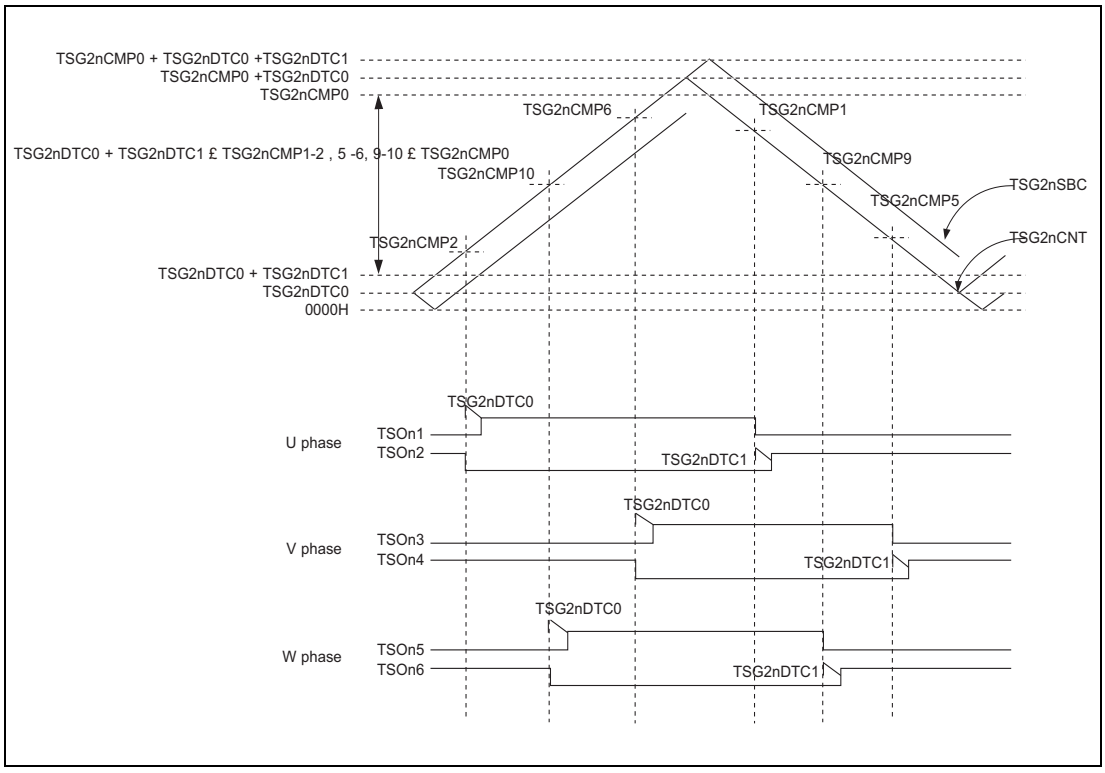
(1) PWM Setting

When a symmetric triangular wave is used, the output control of each of the U phase, V phase, and W phase is done by setting the set timing and the clear timing to the same value to the TSG2nCMPU, TSG2nCMPV, and TSG2nCMPW. When an asymmetric triangular wave is used, the output control of each phase is done by setting TSG2nCMPm as follows (m = 1, 2, 5, 6, 9, 10):

Prerequisites

- The clear timing of PWM of the voltage data signal of U, V and W phases is set with TSG2nCMP1, TSG2nCMP5, and TSG2nCMP9.
- The set timing of PWM of the voltage data signal of U, V, and W phases is set with TSG2nCMP2, TSG2nCMP6, and TSG2nCMP10.
- The set and the clear timings of each phase can also be set with TSG2nCMP1W (TSG2nCMP1 and TSG2nCMP2), TSG2nCMP5W (TSG2nCMP5 and TSG2nCMP6), and TSG2nCMP9W (TSG2nCMP9 and TSG2nCMP10)
- TSG2nCMPm can only be set to an even value (m = 1, 2, 5, 6, 9, 10).

(2) Timer Output



NOTE

When output is controlled by the asymmetric triangular waveform, the following conditions apply to setting of TSG2nCMPm (m = 1, 2, 5, 6, 9, 10):

- TSG2nDTC0 + TSG2nDTC1 ≤ TSG2nCMPm ≤ TSG2nCMP0
- Only when TSG2nCMPm = TSG2nCMP(m+1) or TSG2nCMPm = TSG2nCMP(m+1) + 2, it is possible to set TSG2nCMPm under the condition 0000_H ≤ TSG2nCMPm ≤ TSG2nCMP0 + TSG2nDTC0 + TSG2nDTC1, which also applies to the case in which the symmetric triangular wave is used.

23.11.3 SP-PWM Mode (Shifted-pulse - Pulse Width Modulation Mode)

Overview

In this mode, a 6-phase PWM can be generated using the 16-bit counter and the 16-bit compare registers.

Prerequisites

- The PWM signal cycle is set in TSG2nCMP0.
- The set timings of the U phase, V phase, and W phase are set with TSG2nCMP2, TSG2nCMP6, and TSG2nCMP10, while the clear timings of these phases are set with TSG2nCMP1, TSG2nCMP5, and TSG2nCMP9 (when set timing and clear timing are used for control).
- The set timings of the U phase, V phase, and W phase are set with TSG2nCMP2, TSG2nCMP6, and TSG2nCMP10 while the active periods of these phases are set with TSG2nUPW, TSG2nVPW, and TSG2nWPW.
  The sum of the set values of TSG2nCMP2, TSG2nCMP6, TSG2nCMP10, and the set values of TSG2nUPW, TSG2nVPW, TSG2nWPW are set to TSG2nCMP1, TSG2nCMP5, and TSG2nCMP9 respectively (when set timing and active period are used for control).

Functional description

Set the carrier period and the set timings and duty of U phase, V phase, and W phase. The counting up begins when TSG2nTRG0.TSG2nTS is set to 1.

The 16-bit counter counts up from 0000_H and is cleared by match with TSG2nCMP0.

The dead time is set with TSG2nDTC0 and TSG2nDTC1. TSG2nDTC0 sets the dead time of inverse phase (OFF) to positive phase (ON) switch while TSG2nDTC1 sets that of positive phase (OFF) to inverse phase (ON) switch. The 10-bit counters (TSG2nDTT1 to TSG2nDTT3) for dead time generation load the set values of TSG2nDTC0 and TSG2nDTC1 at compare match of the 16-bit counter with the TSG2nCMPm buffer register (m = 1, 2, 5, 6, 9, 10) and start down-counting.

INTTSG2nIm interrupts (m = 1, 2, 5, 6, 9, 10) are generated by the compare match of the 16-bit counter with the TSG2nCMP1, TSG2nCMP2, TSG2nCMP5, TSG2nCMP6, TSG2nCMP9, and TSG2nCMP10 buffer registers.

NOTES

SP-PWM mode is enabled when TSG2nTRG0.TSG2nMD1 and TSG2nMD0 = 10_B.

23.11.3.1 Basic Timing Chart

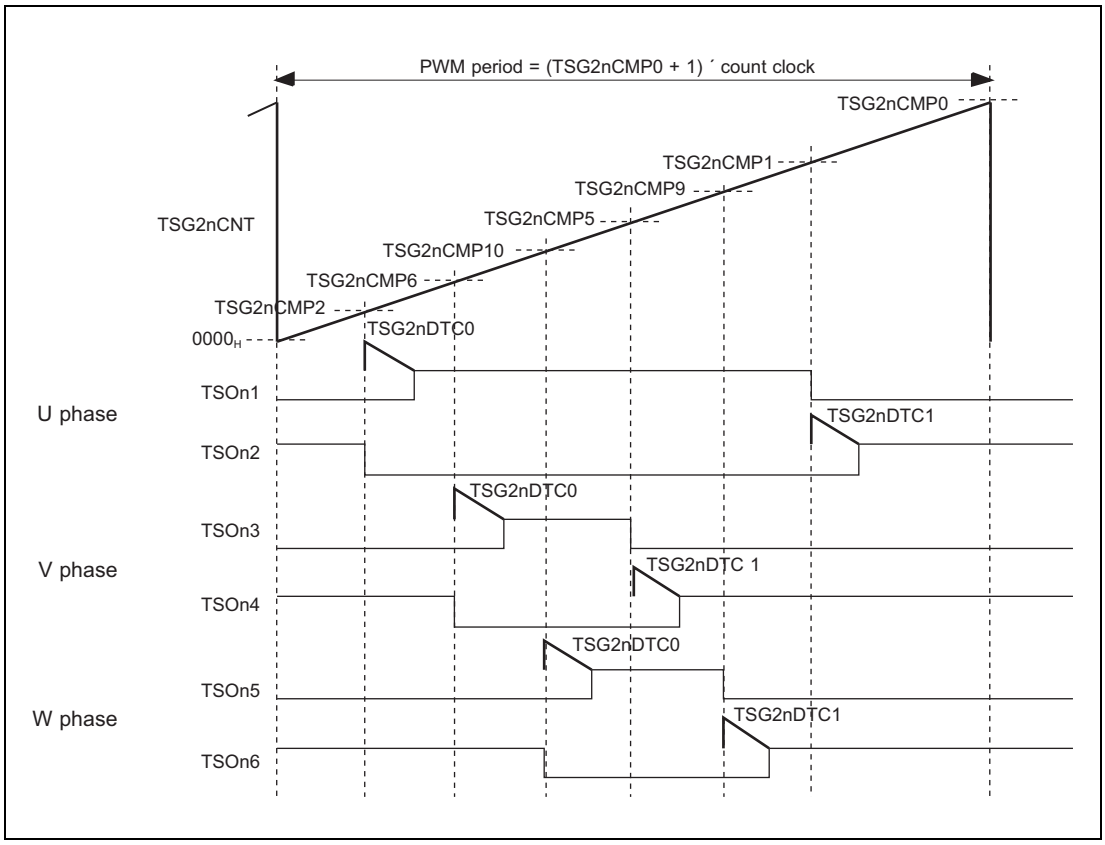


Figure 23.65 Basic Timing in SP-PWM Mode

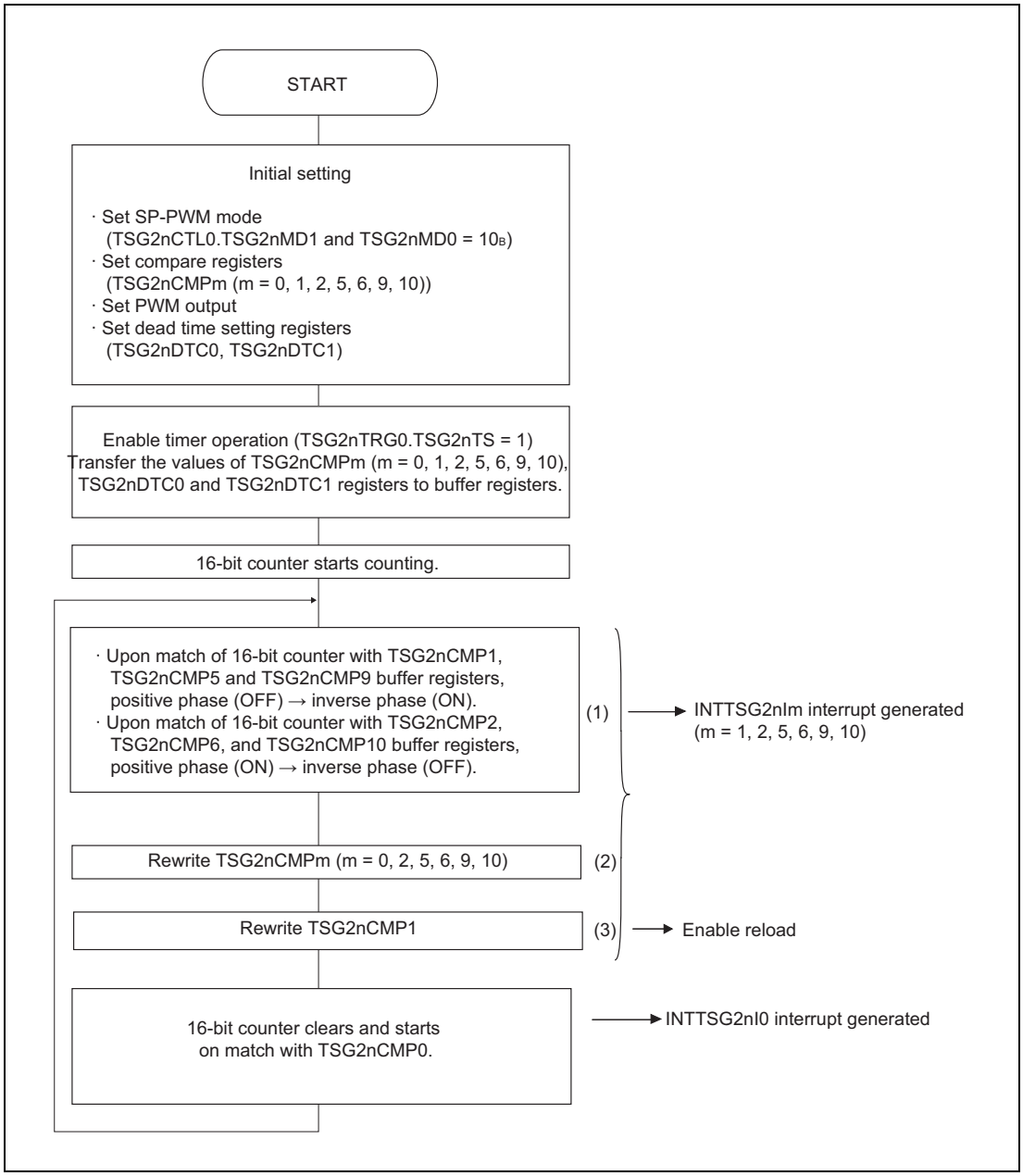


Figure 23.66 Basic Operation Flow in SP-PWM Mode

NOTE

The timing of (1) may be different depending on the rewriting timing of (2) and (3).
Be sure to do step (2) before (3).

23.11.3.2 List of SP-PWM Mode Operations

Table 23.58 Counter Functions in SP-PWM Mode

Operation		Setting Condition
16-bit counter	Start	TSG2nTRG0.TSG2nTS = 0 → 1
	Clear	Compare match of TSG2nCMP0 buffer register with 16-bit counter
	Stop	TSG2nTRG1.TSG2nTT = 0 → 1

Table 23.59 Compare Registers and Dead Time Setting Register Functions in SP-PWM Mode

Register	Rewriting Method	Rewrite during Operation	Function
TSG2nCMP0	Reload/Anytime rewrite	Possible	Setting period
TSG2nUPW	Reload/Anytime rewrite	Possible	PWM control for U phase
TSG2nCMP1W (TSG2nCMP1, TSG2nCMP2)	Reload/Anytime rewrite		
TSG2nVPW	Reload/Anytime rewrite	Possible	PWM control for V phase
TSG2nCMP5W (TSG2nCMP5, TSG2nCMP6)	Reload/Anytime rewrite		
TSG2nWPW,	Reload/Anytime rewrite	Possible	PWM control for W phase
TSG2nCMP9W (TSG2nCMP9, TSG2nCMP10)	Reload/Anytime rewrite		
TSG2nDTC0, TSG2nDTC1	Reload	Possible	Period and dead time
TSG2nDCMP0W, TSG2nDCMP2	Reload/Anytime rewrite	Possible	Diagnostic signal output or A/D conversion trigger

Table 23.60 Output Functions in SP-PWM Mode

Pin	Function
TSOn1	PWM output with dead time by compare match of TSG2nCMP1 buffer register (clear timing) or TSG2nCMP2 buffer register (set timing) with 16-bit counter
TSOn2	Output inverse phase with respect to TSOn1 (with dead time)
TSOn3	PWM output with dead time by compare match of TSG2nCMP5 buffer register (clear timing) or TSG2nCMP6 buffer register (set timing) with 16-bit counter
TSOn4	Output inverse phase with respect to TSOn3 (with dead time)
TSOn5	PWM output with dead time by compare match of the TSG2nCMP9 buffer register (clear timing) or the TSG2nCMP10 buffer register (set timing) with the 16-bit counter
TSOn6	Output inverse phase with respect to TSOn5 (with dead time)
TSOn7	Diagnostic signal output or pulse output by A/D conversion trigger

Table 23.61 Interrupt Requests in SP-PWM Mode

Interrupt	Function
INTTSG2nIm (m = 0, 1, 2, 5, 6, 9, 10)	Compare match of TSG2nCMPm buffer register with 16-bit counter (m = 0, 1, 2, 5, 6, 9, 10)
INTTSG2nIER	Error
INTTSG2nIVLY	—
INTTSG2nIPEK	Peak interrupt (generated at the same timing as INTTSG2nI0)
INTTSG2nIWN	Warning

Table 23.62 Compare Match Timing in SP-PWM Mode

Compare Match	Timing
TSG2nCMP0	When 16-bit counter changes from TSG2nCMP0 to 0000_H
TSG2nCMPm (m = 1, 2, 5, 6, 9, 10)	After match of 16-bit counter and TSG2nCMPm is detected (m = 1, 2, 5, 6, 9, 10)

Table 23.63 Example of Setting Each Timer Output Condition in SP-PWM Mode

			Output Duty	
Pin	**Item**	**Output Period**	**Output Condition**	**Setting Condition**
TSOn1, TSOn3, TSOn5	PWM output	(TSG2nCMP0 + 1) × count clock	Output an inactive level throughout one period (duty 0%)	TSG2nCMPm = TSG2nCMP(m + 1) or TSG2nCMP(m + 1) > TSG2nCMP0 (m = 1, 5, 9)
			Output an active level of one count clock in one period	TSG2nCMPm = TSG2nCMP(m + 1) + 1 TSG2nCMP(m + 1) = TSG2nCMPm - 1 (m = 1, 5, 9)
			Output an inactive level of one count clock in one period	TSG2nCMPm = TSG2nCMP(m + 1) - 1 TSG2nCMP(m + 1) = TSG2nCMPm + 1 (m = 1, 5, 9)
			Output an active level throughout one period (duty 100%)	TSG2nCMPm > TSG2nCMP0 TSG2nCMP(m + 1) ≤ TSG2nCMP0 (m = 1, 5, 9)
TSG2nO2, TSG2nO4, TSG2nO6	PWM output	(TSG2nCMP0 + 1) × count clock	Output an inactive level throughout one period (duty 0%)	TSG2nCMPm = TSG2nCMP(m - 1) or TSG2nCMP(m - 1) > TSG2nCMP0 (m = 2, 6, 10)
			Output an active level of one count clock in one period	TSG2nCMPm = TSG2nCMP(m - 1) - 1 TSG2nCMP(m - 1) = TSG2nCMPm + 1 (m = 2, 6, 10)
			Output an inactive level of one count clock in one period	TSG2nCMPm = TSG2nCMP(m - 1) + 1 TSG2nCMP(m - 1) = TSG2nCMPm - 1 (m = 2, 6, 10)
			Output an active level throughout one period (duty 100%)	TSG2nCMPm > TSG2nCMP0 (m = 2, 6, 10)
TSOn7	Diagnostic signal output or pulse output by A/D conversion trigger	(TSG2nCMP0 + 1) × count clock	Please refer to **Section 23.9, A/D Conversion Trigger Function**.	

23.11.3.3 Various Settings of SP-PWM Mode

Mode Setting

SP-PWM mode is entered by setting TSG2nCTL0.TSG2nMD1-TSG2nMD0 to 10_B.

Setting timer output

The output pins TSOn1-TSOn6 are controlled by setting TSG2nIOC0, TSG2nIOC2, and TSG2nIOC3.

The TSOn7 pin provides output pulse as diagnostic output or analog to digital conversion trigger. The pin should be set as necessary.

Enabling error interrupt generation

Error interrupt (INTTSG2nIER) due to the detection of the simultaneous active state of the positive and inverse phases is enabled by setting TSG2nIOC1.TSG2nEOC to 1. For details, refer to **Section 23.10, Error and Warning Interrupts**.

Setting rewriting timing of register with reload function

With the TSG2nCTL3.TSG2nRMC, reload (simultaneous rewrite) or rewrite (anytime) is specified for the registers with reload function. The default setting is 0 (reload). To reload, set TSG2nCTL4.TSG2nPRE to 1.

No reload timing is generated when TSG2nPRE = 0.

When "anytime rewrite" is specified, the unintended output may be generated depending on the rewrite timing.

Setting analog to digital conversion trigger output

The analog to digital conversion trigger 0 (TSTADT0 signal) is set with TSG2nCTL5.TSG2nAT09 to TSG2nAT00.

TSG2nAT09 to TSG2nAT00 is used to enable or disable the A/D conversion trigger output on timing match of TSG2nDCMP2 to TSG2nDCMP0 with the 16-bit counter (up count).

TSG2nCTL6.TSG2nAT19 to TSG2nAT10 is used to set the analog to digital conversion trigger 1 (TSTADT1 signal).

To set the match timing of the 16-bit counter and TSG2nDCMP2 to TSG2nDCMP0, set the compare value to each register.

The skipping function can be used for TSTADT0 and TSTADT1 signals. TSG2nACC00 and TSG2nACC01 of TSG2nCTL5, and TSG2nACC10 and TSG2nACC11 of TSG2nCTL6 can be used to select the skipping rate among 1/1, 1/2, 1/4, and 1/8.

CAUTIONS

- **Be sure to set TSG2nCTL5, TSG2nCTL6, and TSG2nDCMP2 to TSG2nDCMP0 correctly when the timing pulse of analog to digital conversion trigger is output to TSOn7.**
- **In SP-PWM mode, no trough interrupt (INTTSG2nIVLY) is generated. Therefore, TSG2nCTL5.TSG2nAT00 and TSG2nCTL6.TSG2nAT10 must be set to 0.**
- **In SP-PWM mode, the 16-bit sub-counter does not operate. Therefore, TSG2nAT09, and TSG2nAT08 in TSG2nCTL5, and TSG2nAT19, and TSG2nAT18 in TSG2nCTL6 must be set to 0.**
- **In SP-PWM mode, down counting by the 16-bit counter is not generated. Therefore, TSG2nAT07, TSG2nAT05, and TSG2nAT03 in TSG2nCTL5 and TSG2nAT17, TSG2nAT15, and TSG2nAT13 in TSG2nCTL6 should be set to 0.**

Setting dead time

The dead time can be set with TSG2nDTC0 and TSG2nDTC1.

The dead time is calculated by the following expressions:

PCLK × TSG2nDTC0

PCLK × TSG2nDTC1

TSG2nDTC0 can set the time between a change of TSOn2, TSOn4, and TSOn6 to the inactive state and a change of TSOn1, TSOn3, and TSOn5 to the active state, respectively.

TSG2nDTC1 can set the time between a change of TSOn1, TSOn3, and TSOn5 to the inactive state, and a change of TSOn2, TSOn4, and TSOn6 to the active state, respectively.

Carrier period

Set the carrier period with TSG2nCMP0 according to the following expression:

TSG2nCMP0 = (carrier period/count clock period) - 1

CAUTION

PWM output with 100% duty cannot be produced when TSG2nCMP0 = $FFFF_H$.

Setting duty (PWM width)

The duty of U phase, V phase, and W phase is set with TSG2nCMPm, TSG2nUPW, TSG2nVPW, and TSG2nWPW (m = 1, 2, 5, 6, 9, 10), respectively.

- The set timings of the U phase, V phase, and W phase are set with TSG2nCMP2, TSG2nCMP6, and TSG2nCMP10, and the clear timings are set with TSG2nCMP1, TSG2nCMP5, and TSG2nCMP9. (The set and clear timing setting is used for control.)
- The set timings of U phase, V phase, and W phase is set with TSG2nCMP2, TSG2nCMP6, and TSG2nCMP10 while the active periods of these phases are set with TSG2nUPW, TSG2nVPW, and TSG2nWPW.
  The sum of the set values of TSG2nCMP2, TSG2nCMP6, and TSG2nCMP10, and the set values of TSG2nUPW, TSG2nVPW, and TSG2nWPW are set to TSG2nCMP1, TSG2nCMP5, and TSG2nCMP9 respectively (when set timing and active period are used for control).

23.11.3.4 Dead Time Control in SP-PWM mode

Duty setting registers are TSG2nCMPm (m = 1, 2, 5, 6, 9, 10), TSG2nUPW, TSG2nVPW, and TSG2nWPW and register for setting the period is TSG2nCMP0. The 6-phase PWM waveform of a variable duty is output by using these registers. To achieve the dead time control, there are six 10-bit down counters that operate synchronously with the count clock of the 16-bit counter and two dead time setting registers (TSG2nDTC0 and TSG2nDTC1). TSG2nDTC0 is used for setting a dead time from a change of the inverse phase to the inactive state to a change of the positive phase to the active state. TSG2nDTC1 is used for setting a dead time from a change of the positive phase to the inactive state to a change of the inverse phase to the active state. **Figure 23.67** shows an example of the output waveform.

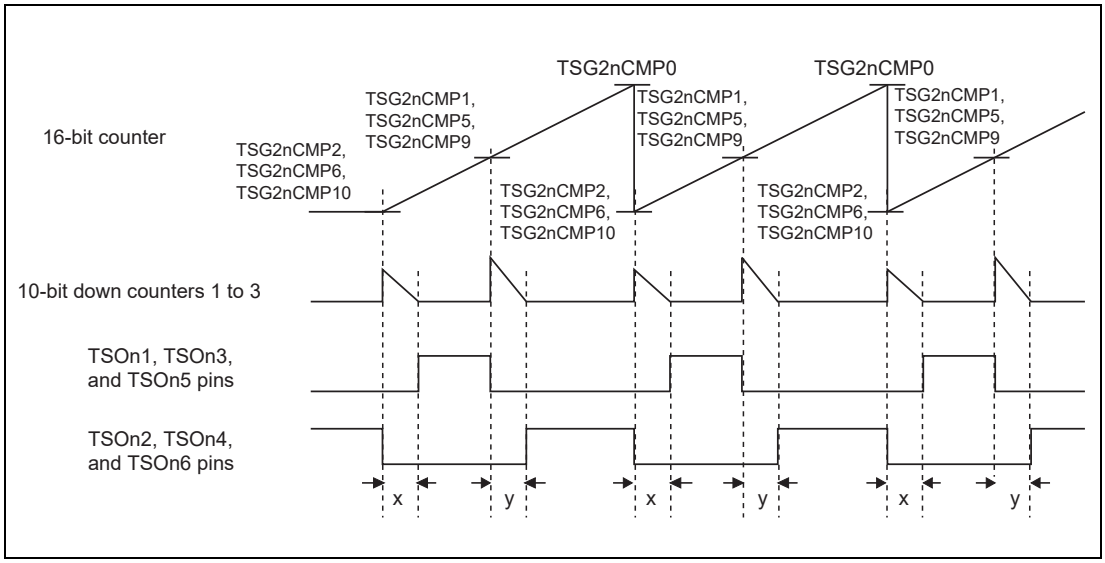


Figure 23.67 Example of Output Waveform in SP-PWM Mode

NOTE

x : TSG2nDTC0 register values

y : TSG2nDTC1 register values

23.11.3.5 Software Output Control Function in SP-PWM Mode

TSG2nOPT0.TSG2nSOC, TSG2nIDC, and TSG2nOPT1.TSG2nSPC2 to TSG2nSPC0 are used to control timer output by software.

As shown in **Figure 23.68**, the output control is switched immediately when TSG2nSOC is set to 1. If the dead time is set, the period of the dead time is guaranteed. After that, when TSG2nSOC is set to 0, output control is retained. When the reload timing is generated, output control is switched to SP-PWM mode output control.

For details, refer to **Section 23.11.5, Software Output Control Function**.

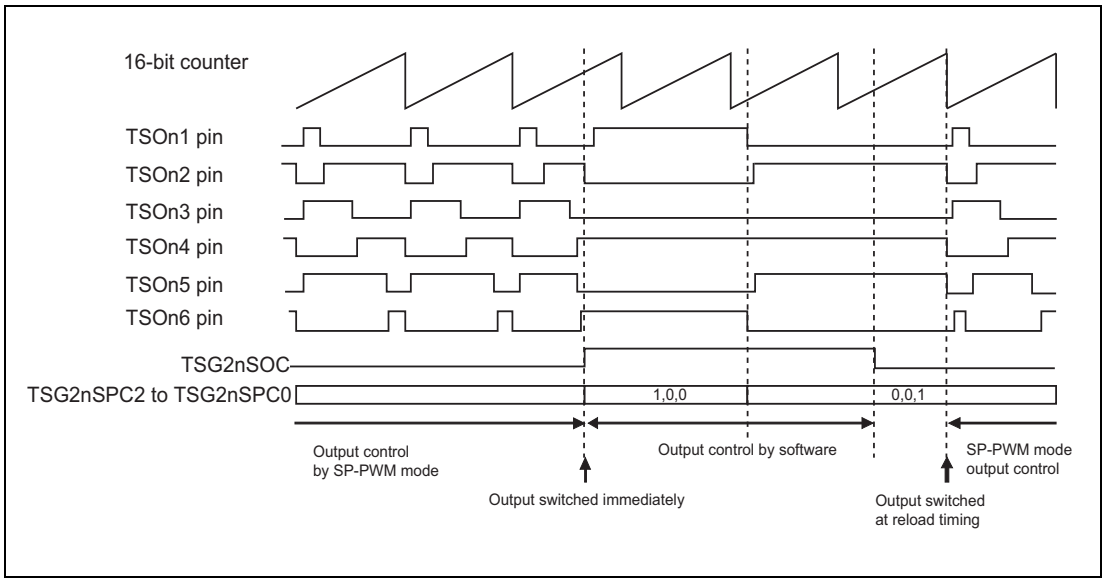


Figure 23.68 Example of Output Control Switching from SP-PWM Mode Control to Software Control

(1) Procedure on Software Output Control Processing

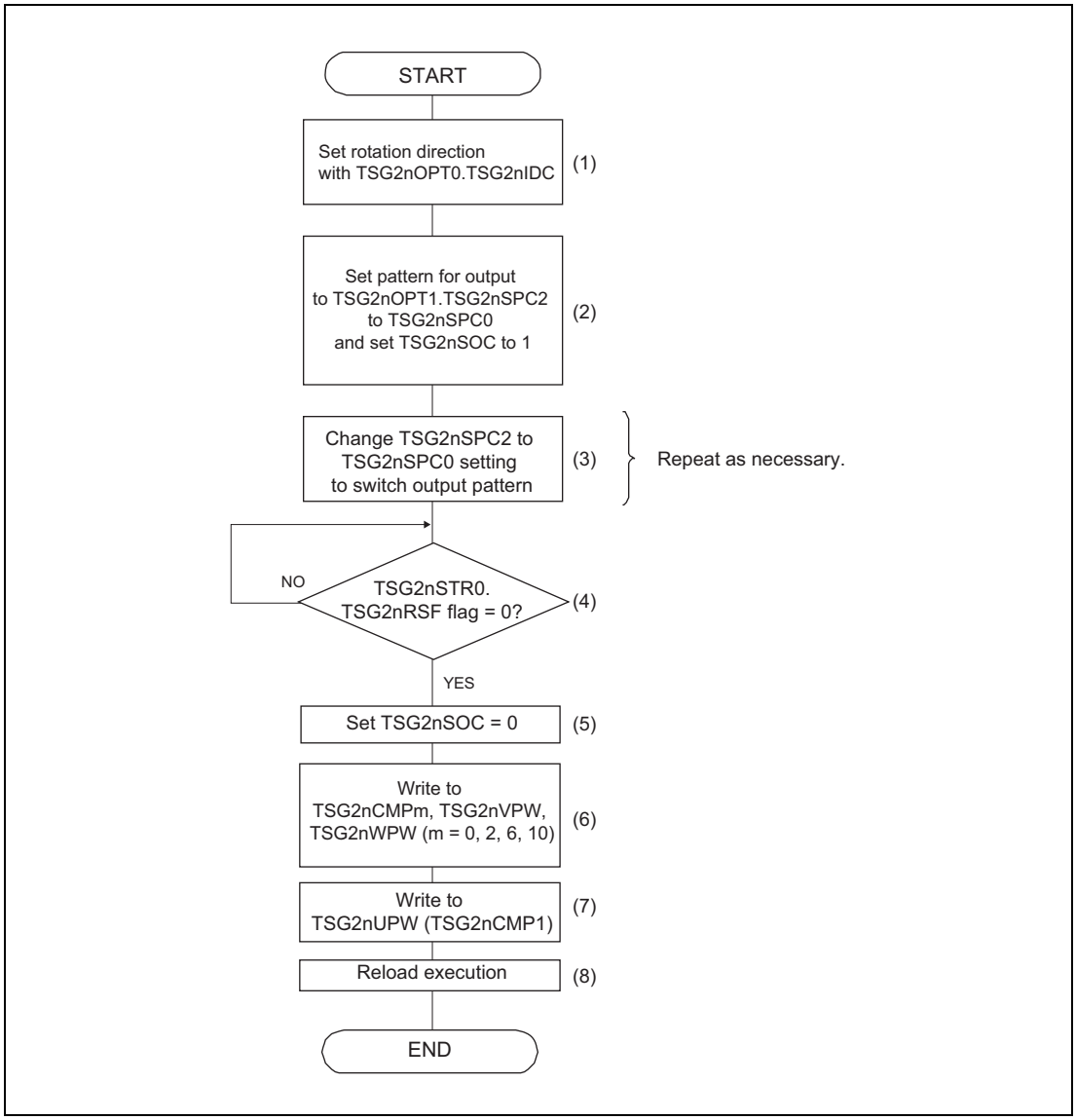


Figure 23.69 Flow of Software Output Control

The procedure for software output control is described below.

(1) Set TSG2nOPT0 to determine the rotation direction. The timer output has a 180-degree phase shift between when TSG2nIDC = 0 and when TSG2nIDC = 1. With the software output control function, the timer output will not change only by rewriting the TSG2nOPT0 bit.

(2) Set the pattern for output to TSG2nOPT1. At the same time, set TSG2nOPT0.TSG2nSOC to 1 to switch to software output control mode.

(3) Change the output pattern setting for TSG2nSPC2-TSG2nSPC0 to change the timer output. During software control mode, the following registers can be modified: TSG2nTRG0.TSG2nTS, TSG2nCTL3-TSG2nCTL6, TSG2nOPT0, TSG2nOPT1, TSG2nCMP0-TSG2nCMP12, TSG2nDTC0, and TSG2nDTC1.

(4) Ensure that the reload request flag TSG2nSTR0.TSG2nRSF is 0. If TSG2nRSF is 1, do not shift to the following procedure until it goes 0.

(5) By clearing TSG2nSOC to 0, software control starts to be released (not yet released here).

(6) After software output control is released, set the compare registers if necessary. Move to the following procedure if no setting is required. In addition, change the registers with the reload function if necessary.

(7) Write TSG2nUPW (TSG2nCMP1) to start reloading.

(8) Reload is executed and software output control is released.

CAUTION

Be sure to execute reload after execution of steps (4), (5), (6), and (7). Unless reload can be executed, software output control cannot be released

23.11.4 120-DC Mode

Overview

In this mode, PWM output period set to TSG2nCMP0 and timer output (TSOn1 to TSOn6) according to the duty cycle set to TSG2nCMP1 to TSG2nCMP12 are controlled with two types of pattern inputs (software output control method and pattern switch method) to perform 120-DC control.

Prerequisites

- Set the PWM period to TSG2nCMP0.
- Set the PWM duty to TSG2nCMP1 to TSG2nCMP12 and set the output pattern to TSG2nPAT0W and TSG2nPAT1W.

Functional description

Set the PWM period, set the duty cycle to individual compare register, and set the pattern to be output to the pattern register. Setting TSG2nTRG0.TSG2nTS to 1 starts counting.

The 16-bit counter starts counting from 0000_H, and is cleared by the match with TSG2nCMP0.

INTTSG2nI1 to INTTSG2nI12 interrupts are generated by a compare match of the 16-bit counter and TSG2nCMP1 to TSG2nCMP12 buffer registers, respectively.

NOTE

120-DC mode is valid when TSG2nCTL0.TSG2nMD1 to TSG2nMD0 are set to 11_B.

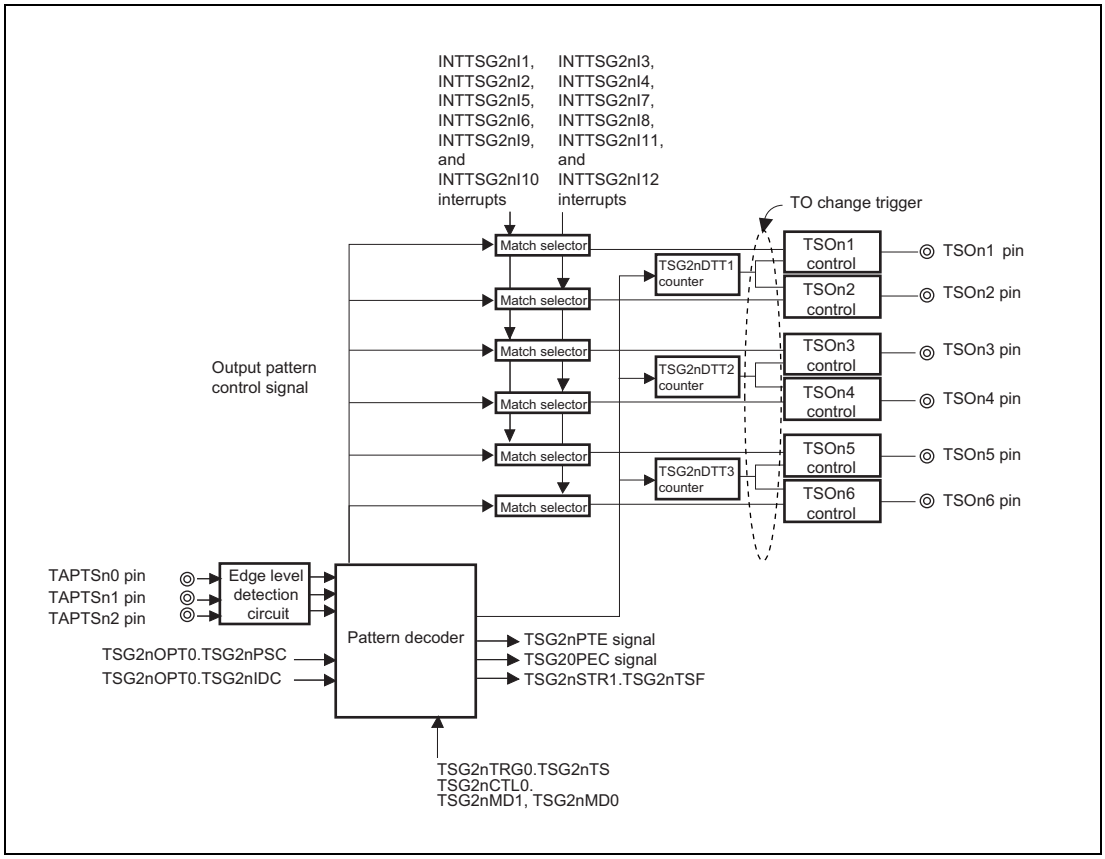
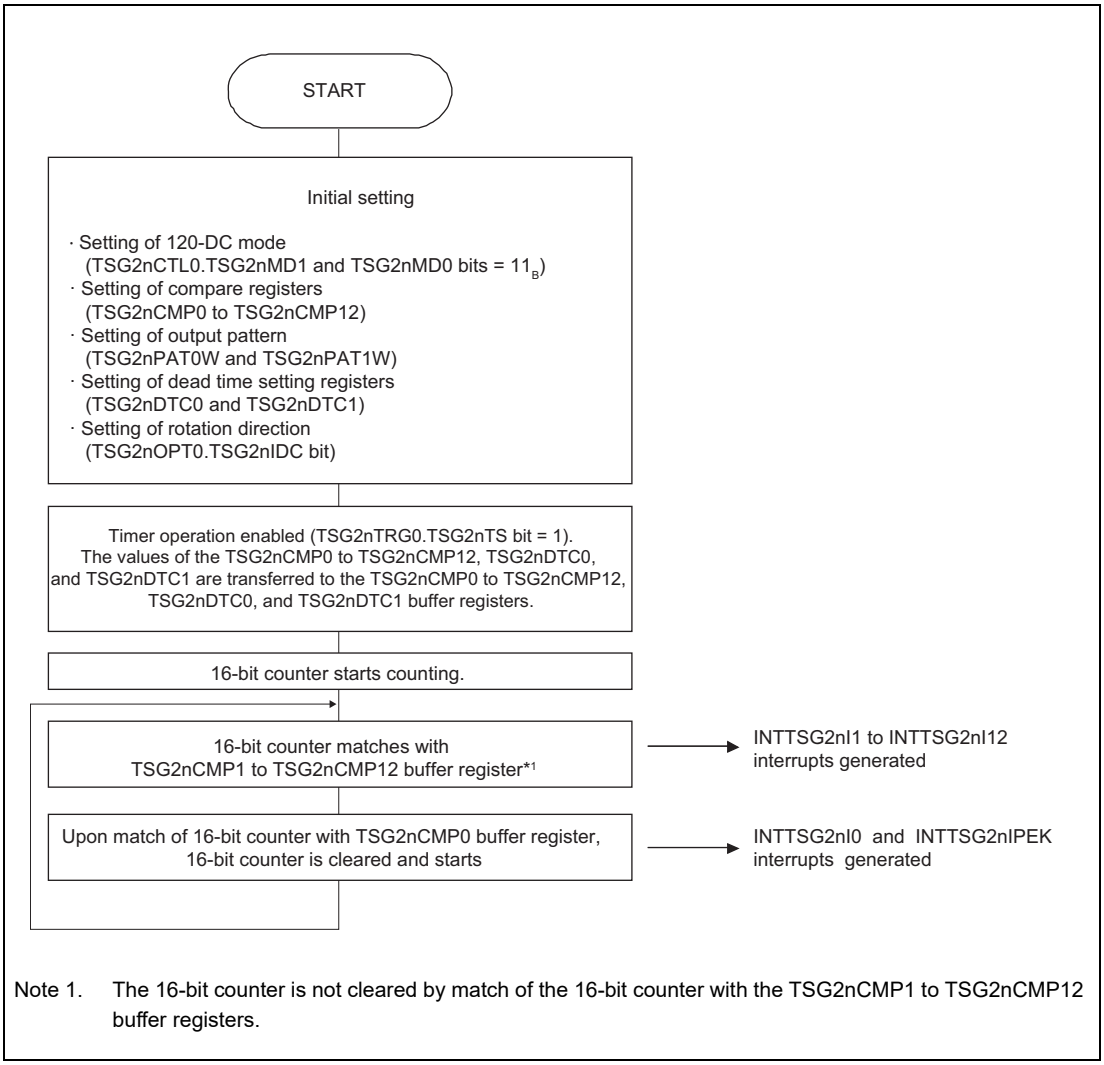


Figure 23.70 Block Diagram in 120-DC Mode



Note 1. The 16-bit counter is not cleared by match of the 16-bit counter with the TSG2nCMP1 to TSG2nCMP12 buffer registers.

Figure 23.71 Basic Operation Flow in 120-DC Mode

23.11.4.1 List of Operations in 120-DC Mode

Table 23.64 Counter Functions in 120-DC Mode

Operation		Setting Condition
16-bit counter	Start	TSG2nTRG0.TSG2nTS = 0 → 1
	Clear	Match of TSG2nCMP0 value and 16-bit counter value, or output pattern switch timing
	Stop	TSG2nTRG1.TSG2nTT = 0 → 1

Table 23.65 Functions of Compare Registers and Dead Time Setting Registers in 120-DC Mode

Register	Rewrite Method	Rewrite during Operation	Function
TSG2nCMP0	Reload	Possible	Setting period
TSG2nCMPm (m = 1 to 12)	Reload	Possible	Setting PWM duty
TSG2nDCMP0W, TSG2nDCMP2	Reload	Possible	Outputting diagnostic signal or setting A/D conversion trigger
TSG2nDTC0, TSG2nDTC1	Reload	Possible	Setting dead time

Table 23.66 Timer Input Function in 120-DC Mode

Pin/Signal	Function
TAPTSn2 to TAPTSn0 pins	Pattern input (3 phases)

Table 23.67 Timer Output Function in 120-DC Mode

Pin/Signal	Function
TSOn1 pin	PWM output (with dead time) by compare match of the TSG2nCMPm buffer register (m = 1, 2, 5, 6, 9, 10) with the 16-bit counter and by selecting output pattern through TSG2nPAT0W setting.
TSOn2 pin	PWM output (with dead time) by compare match of the TSG2nCMPm buffer register (m = 3, 4, 7, 8, 11, 12) and the 16-bit counter and by selecting output pattern through TSG2nPAT1W setting.
TSOn3 pin	PWM output (with dead time) by compare match of the TSG2nCMPm buffer register (m = 1, 2, 5, 6, 9, 10) and the 16-bit counter and by selecting output pattern through TSG2nPAT0W setting.
TSOn4 pin	PWM output (with dead time) by compare match of the TSG2nCMPm buffer register (m = 3, 4, 7, 8, 11, 12) and the 16-bit counter and by selecting output pattern through TSG2nPAT1W setting.
TSOn5 pin	PWM output (with dead time) by compare match of the TSG2nCMPm buffer register (m = 1, 2, 5, 6, 9, 10) and the 16-bit counter and by selecting output pattern through TSG2nPAT0W setting.
TSOn6 pin	PWM output (with dead time) by a compare match of the TSG2nCMPm buffer register (m = 3, 4, 7, 8, 11, 12) and the 16-bit counter and by selecting output pattern through TSG2nPAT1W setting.
TSOn7 pin	Diagnostic signal output or pulse output by A/D conversion trigger
TSG2nPTE signal	Toggle signal by change in input pattern

Table 23.68 Interrupt Requests in 120-DC Mode

Interrupt	Function
INTTSG2nIm (m = 0 to 12)	Compare match of TSG2nCMPm buffer register and 16-bit counter (m = 0 to 12)
INTTSG2nIER	Error
INTTSG2nIVLY	—
INTTSG2nIPEK	Peak interrupt (generated at the same timing as INTTSG2nI0)
INTTSG2nIWN	Warning interrupt

Table 23.69 Compare Match Timing in 120-DC Mode

Compare Match	Timing
TSG2nCMP0	When 16-bit counter changes from TSG2nCMP0 to 0000_H
TSG2nCMPm (m = 1 to 12)	After detecting the match of 16-bit counter and TSG2nCMPm (m = 1 to 12)

Table 23.70 Example of Setting Each Timer Output Condition in 120-DC Mode

			Output Duty	
Pin	Item	Output Period	Output Condition	Setting Condition
TSOnm (m = 1 to 6)	PWM output	(TSG2nCMP0 + 1) × count clock	See **23.11.4.6, List of Output Patterns in 120-DC Mode**.	—
TSOn7	Diagnostic signal output or pulse output by A/D conversion trigger	(TSG2nCMP0 + 1) × count clock	See **23.9, A/D Conversion Trigger Function**.	—

23.11.4.2 Various Settings of 120-DC Mode

Mode setting

120-DC mode can be used by setting TSG2nCTL0.TSG2nMD1 and TSG2nMD0 are set to 11_B.

Setting timer output

The output pins TSOn1 to TSOn6 are controlled by setting TSG2nIOC0, TSG2nIOC2, and TSG2nIOC3.

The TSOn7 pin outputs pulses of the diagnostic output or the A/D conversion trigger. Please set it as required.

Enabling error interrupt generation

With TSG2nIOC1.TSG2nEOC = 1, the error interrupt (INTTSG2nIER) generation is enabled when the simultaneous active state of the positive phase and inverse phase is detected. For details, see **Section 23.10, Error and Warning Interrupts**.

Setting register rewrite timing

Reloading the registers with the reload function is activated with TSG2nCTL3.TSG2nRMC (simultaneous rewrite; default setting is 0 = reload). Set TSG2nCTL4.TSG2nPRE to 1 when reload is used.

The reload timing is not generated if TSG2nPRE is 0.

Setting A/D conversion trigger output

To set A/D conversion trigger 0 (TSG2nADTRG0 signal), use TSG2nCTL5.TSG2nAT09 to TSG2nAT00.

With TSG2nAT09 to TSG2nAT00, A/D conversion trigger output is enabled or disabled at the match of the 16-bit counter and TSG2nDCMP2 to TSG2nDCMP0 (during up count).

To set A/D conversion trigger 1 (TSG2nADTRG1 signal), use TSG2nCTL6.TSG2nAT19 to TSG2nAT10.

To set the match timing of the 16-bit counter and TSG2nDCMP2 to TSG2nDCMP0, set the compare value to the pertinent register.

The skipping function can be used for TSG2nADTRG0 and TSG2nADTRG1 signals. Use TSG2nACC01, TSG2nACC00 of TSG2nCTL5 and TSG2nACC11, and TSG2nACC10 in TSG2nCTL6 to select the skipping rate among 1/1, 1/2, 1/4, and 1/8.

CAUTIONS

- **Set TSG2nCTL5, TSG2nCTL6, and TSG2nDCMP2 to TSG2nDCMP0 correctly when using the TSOn7 output for the A/D conversion trigger timing pulse.**
- **In 120-DC mode, a trough interrupt (INTTSG2nIVLY) is not generated. Therefore, set TSG2nAT00 and TSG2nAT10 in TSG2nCTL5 and TSG2nCTL6 to 0.**
- **In 120-DC mode, the 16-bit sub-counter does not operate. Therefore, set TSG2nAT09, TSG2nAT08, TSG2nAT19, and TSG2nAT18 in TSG2nCTL5 and TSG2nCTL6 to 0.**
- **In 120-DC mode, the 16-bit counter does not decrement. Therefore, set TSG2nAT07, TSG2nAT05, TSG2nAT03, TSG2nAT17, TSG2nAT15, and TSG2nAT13 in TSG2nCTL5 and TSG2nCTL6 to 0.**

Setting dead time

The dead time can be set with TSG2nDTC0 and TSG2nDTC1.

The dead time is calculated by the following expressions:

PCLK × TSG2nDTC0

PCLK × TSG2nDTC1

TSG2nDTC0 can set the time between a change of TSOn2, TSOn4, and TSOn6 to the inactive state and a change of TSOn1, TSOn3, and TSOn5 to the active state, respectively.

TSG2nDTC0 can set the time between a change of TSOn1, TSOn3, and TSOn5 to the inactive state to a change of TSOn2, TSOn4, and TSOn6 to the active state.

Carrier period

Set the carrier period with TSG2nCMP0 according to the following expression:

TSG2nCMP0 = (carrier period/count clock period) - 1

Duty (PWM width) setting

The duty of PWM output is set with TSG2nCMP1 to TSG2nCMP12. The setting range of the compare registers is as follows:

$0000_H \leq$ TSG2nCMPm $\leq$ TSG2nCMP0 + 1

CAUTION

Do not set TSG2nCMPm to TSG2nCMP0 + 1 (m = 1 to 12) only when TSG2nCMP0 + 1 < TSG2nCMPm and TSG2nCMP0 = $FFFF_H$.

Output PWM setting

In 120-DC mode, the output pins TSOn1, TSOn3, and TSOn5 are controlled by TSG2nCMP1, TSG2nCMP2, TSG2nCMP5, TSG2nCMP6, TSG2nCMP9, and TSG2nCMP10, and the output pins TSOn2, TSOn4, TSOn6 are controlled by TSG2nCMP3, TSG2nCMP4, TSG2nCMP7, TSG2nCMP8, TSG2nCMP11, and TSG2nCMP12. The duty cycle of a PWM period (TSG2nCMP0) can be set with TSG2nCMP1 to TSG2nCMP12. Setting TSG2nCMP1 to TSG2nCMP12 to 0000H sets the PWM duty cycle to 0%. Setting TSG2nCMP1 to TSG2nCMP12 to TSG2nCMP0 + 1 value sets the PWM duty cycle to 100%. This allows chopping output control and rectangular wave output control.

23.11.4.3 Control Methods in 120-DC Mode

Control methods in 120-DC mode are listed below.

Control Method	Function
Software output control method	Switches the output pattern according to the TSG2nOPT1.TSG2nSPC2 to TSG2nSPC0 setting made by software.
Pattern switch method	Directly switches the output pattern by the pattern input signal of TAPTSn0 to TAPTSn2.

Setting software output control method

Setting TSG2nOPT0.TSG2nSTE = 0 switches the output pattern by software output control. The TSOn1 to TSOn6 pin output is switched according to TSG2nOPT1.TSG2nSPC2 to TSG2nSPC0.

The output order at the beginning of operation is set with TSG2nOPT0.TSG2nIDC. The output pattern is set with TSG2nOPT0.TSG2nPSC.

Operation of software output control method

The PWM output of TSOn1 to TSOn6 pins (PWM output defined by TSG2nCMP1 to TSG2nCMP12) is selected by TSG2nOPT1.TSG2nSPC2 to TSG2nSPC0 by software. To control the dead time, the dead time counter is activated at the falling edge of the signals in each phase and the dead time is inserted.

The 16-bit counter counts based on the carrier period set in TSG2nCMP0. The 16-bit counter is cleared by match of the 16-bit counter and TSG2nCMP0 or by a write access to TSG2nOPT1.TSG2nSPC2 to TSG2nSPC0.

In this method, the pattern is output, which is decoded using information on the output pattern (TSG2nSPC2 to TSG2nSPC0), the electric current direction control bit (TSG2nOPT0.TSG2nIDC), and the order direction control bit (TSG2nOPT0.TSG2nPSC). **Figure 23.91** shows the timer output when the output pattern is changed by software output control.

Immediately after the operation starts (TSG2nTRG0.TSG2nTS = 1), the output pattern of TSG2nSPC2 to TSG2nSPC0 and the pattern set with TSG2nIDC and TSG2nPSC (TSG2nOPT0.TSG2nPSS = 1) are output.

Setting pattern switch method

Setting TSG2nOPT0.TSG2nSTE to 1 and TSG2nPOT to 0 selects the pattern switch method. The TSOn1 to TSOn6 pin output is changed at the change timing of the TAPTSn2 to TAPTSn0 pins.

The output order at the beginning of operation is set with TSG2nOPT0.TSG2nIDC. The initial output pattern is set with TSG2nOPT0.TSG2nPSC. However, after determining the rotation direction (after the value is set to TSG2nSTR1.TSG2nTSF), the setting of TSG2nPSC is disabled.

Operation of pattern switch method

After level detection is performed for the pins (three inputs from the hall sensor), the level-detected signals are decoded. From the decoding result, the PWM output of TSOn1 to TSOn6 pins (PWM output defined by TSG2nCMP1 to TSG2nCMP12) is selected. To control the dead time, the dead time counter is activated at the falling timing of signals in each phase and the dead time is inserted.

The 16-bit counter counts based on the carrier period set in TSG2nCMP0. The 16-bit counter is cleared by match of the 16-bit counter and TSG2nCMP0 or by a change of the input pattern (TAPTSn2 to TAPTSn0 pins).

In this method, the pattern, which is decoded by using information on input pattern (TAPTSn2 to TAPTSn0), the electric current direction control bit (TSG2nOPT0.TSG2nIDC), and the order direction control bit (TSG2nOPT0.TSG2nPSC), is output. **Figure 23.74** to **Figure 23.77** show the timer output when TAPTSn2 to TAPTSn0 pin inputs change. If input pattern 1 is switched to input pattern 4 due to an abnormal input pattern, the output pattern is switched to the pattern corresponding to the input pattern.

Immediately after the operation starts (TSG2nTRG0.TSG2nTS = 1), the output pattern set with the input level of TAPTSn2 to TAPTSn0 pins, TSG2nIDC, and TSG2nPSC (TSG2nOPT0.TSG2nPSS = 1) is output. After the TSG2nTSF value is determined, the output pattern is determined by TSG2nTSF instead of TSG2nPSC.

CAUTION

When connecting the three-phase pulse input signal to the TAPTSn2 to TAPTSn0 pins, confirm that the three-phase pulse input value and the patterns output from the TSOn1 to TSOn6 pins satisfy the expected conditions.
If the expected conditions are not satisfied, change the connection between the three-phase pulse input signal and the TAPTSn2 to TAPTSn0 pins.

23.11.4.4 Timer Output in 120-DC Mode

In 120-DC mode, the PWM output is controlled with TSG2nPAT0W, TSG2nPAT1W, and TSG2nCMP1 to TSG2nCMP12. TSG2nPAT0W, TSG2nCMP1, TSG2nCMP2, TSG2nCMP5, TSG2nCMP6, TSG2nCMP9, and TSG2nCMP10 are set to control the output of TSOn1, TSOn3, and TSOn5 pins. TSG2nPAT1W, TSG2nCMP3, TSG2nCMP4, TSG2nCMP7, TSG2nCMP8, TSG2nCMP11, and TSG2nCMP12 are set with the output of TSOn2, TSOn4, and TSOn6 pins.

With PWM output control, eight types of output patterns can be selected for each of TSOn1, TSOn3, and TSOn5 pins and TSOn2, TSOn4, and TSOn6 pins.

Table 23.71 TSG2nPAT0W Set Value and Output Control

PATmT Value	Output Control
000	Fixed to low
001	PWM output set with TSG2nCMP1
010	PWM output set with TSG2nCMP2
011	PWM output set with TSG2nCMP5
100	PWM output set with TSG2nCMP6
101	PWM output set with TSG2nCMP9
110	PWM output set with TSG2nCMP10
111	Fixed to high

(m = 0, 1, 2, 3, 4, 5)

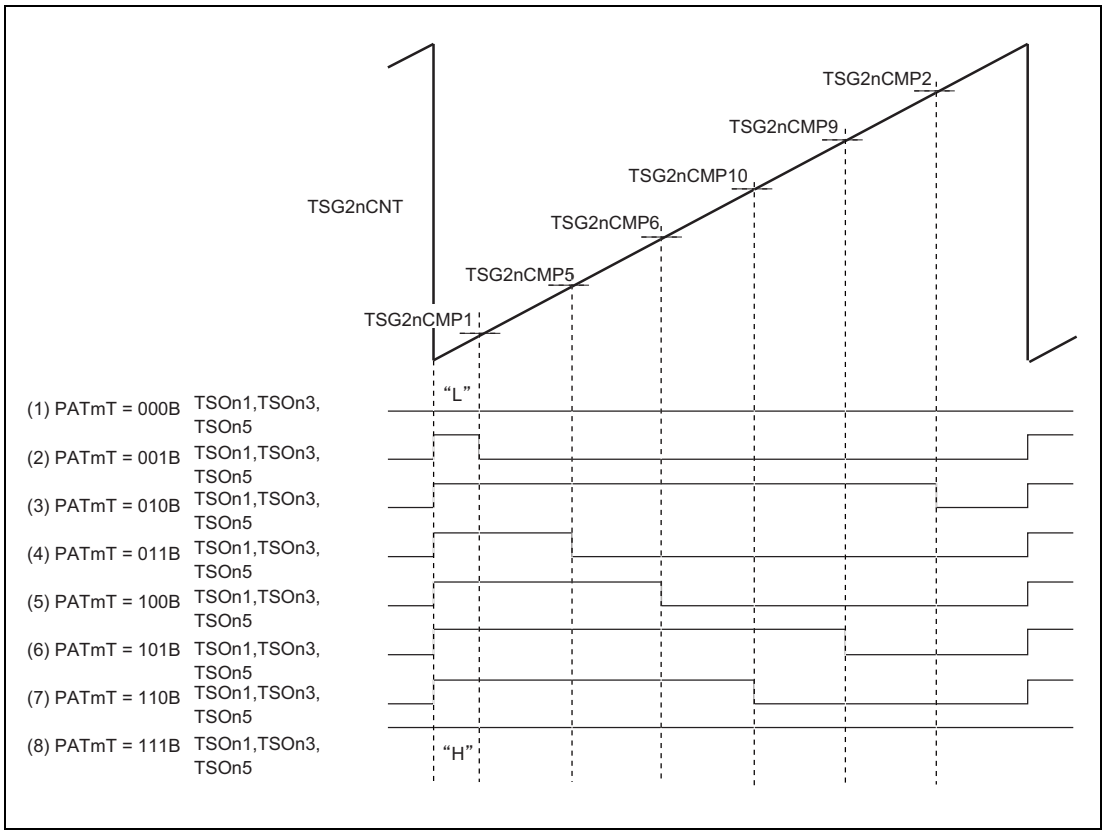


Figure 23.72 TSOn1, TSOn3, TSOn5 Pin Output of Each Output Pattern

Table 23.72 TSG2nPAT1W Set Value and Output Control

PATmB Value	Output Control
000	Fixed to low
001	PWM output set with TSG2nCMP3
010	PWM output set with TSG2nCMP4
011	PWM output set with TSG2nCMP7
100	PWM output set with TSG2nCMP8
101	PWM output set with TSG2nCMP11
110	PWM output set with TSG2nCMP12
111	Fixed to high

(m = 0, 1, 2, 3, 4, 5)

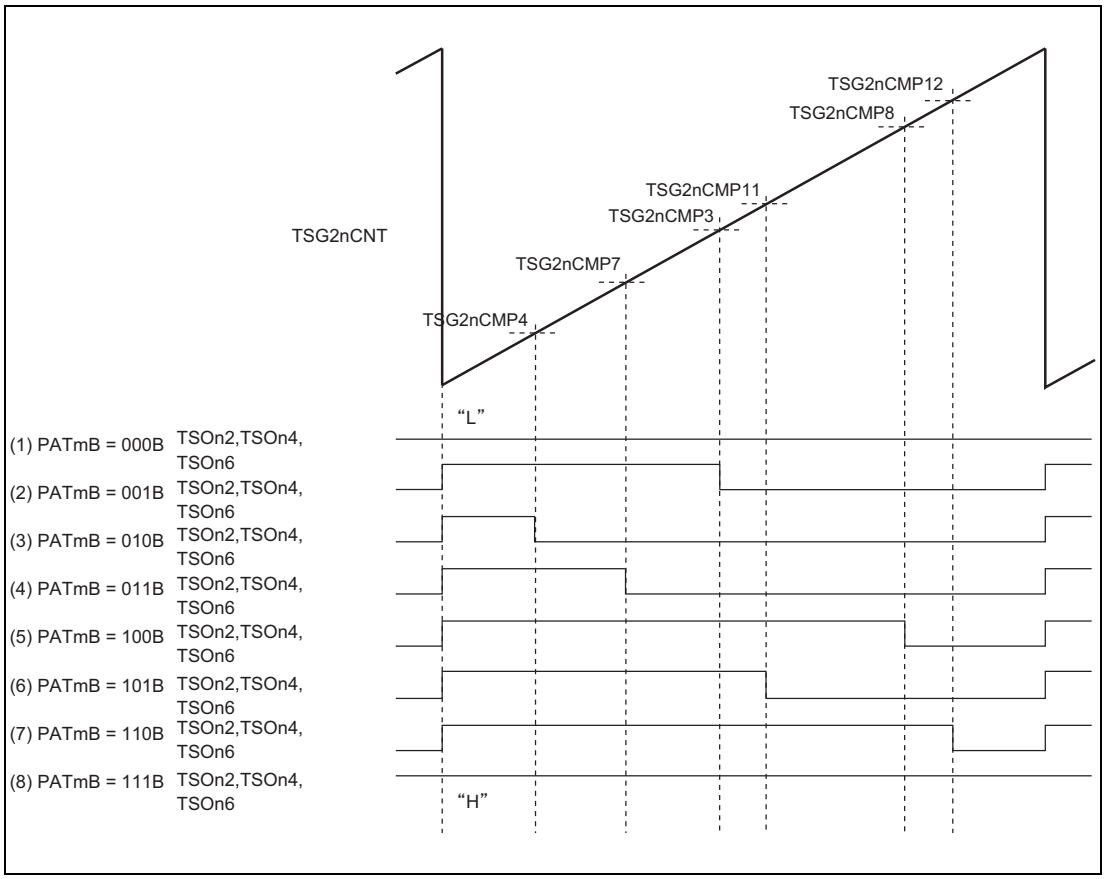


Figure 23.73 TSOn2, TSOn4, TSOn6 Pin Output of Each Output Pattern

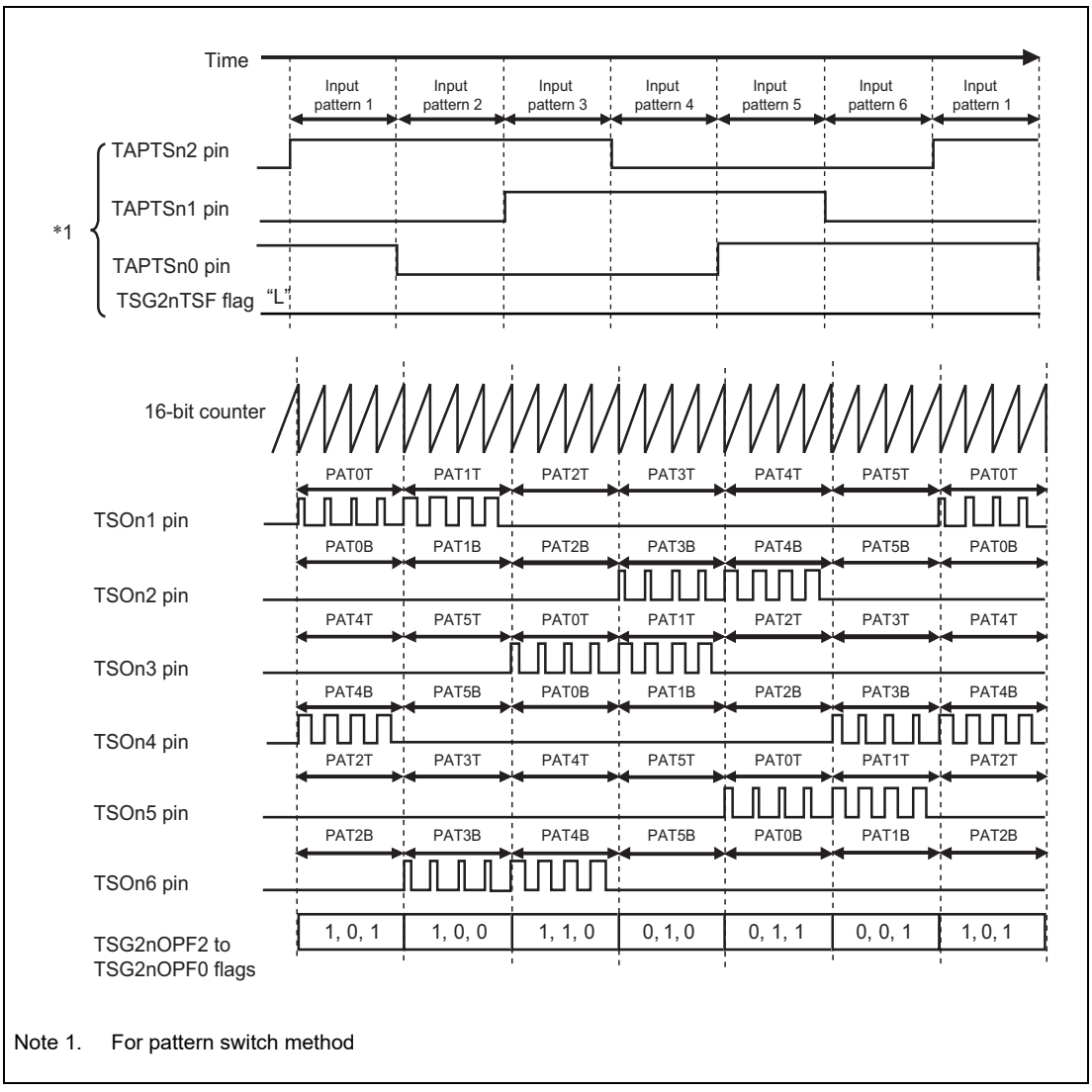
23.11.4.5 Operation in 120-DC Mode

Figure 23.74 to **Figure 23.77** show examples of operation in 120-DC mode.

The TSOn1 to TSOn6 pins detect the input level change of the TAPTSn2 to TAPTSn0 pins, then change the output pattern. The 16-bit counter produces sawtooth waveform, and TSG2nCMP0 to TSG2nCMP12 output PWM signal. The 16-bit counter is cleared to 0000_H each time the counter value matches with TSG2nCMP0 or a change in the TAPTSn2 to TAPTSn0 pins is detected. The timer output pattern is switched each time a change in the TAPTSn2 to TAPTSn0 pins is detected.

NOTE

PAT0T to PAT5T and PAT0B to PAT5B show PWM operation set by TSG2nCMP1 to TSG2nCMP12, respectively.

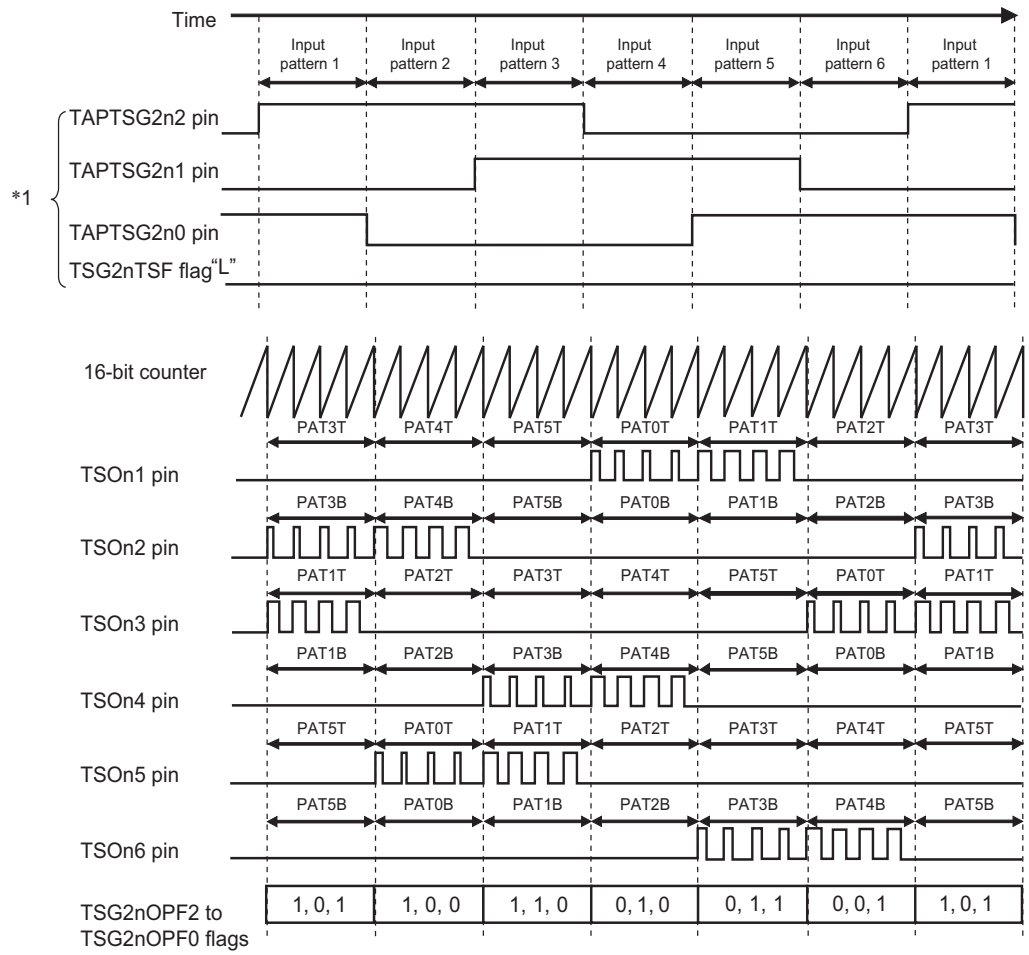


Note 1. For pattern switch method

Figure 23.74 Example of Operation in 120-DC Mode (Normal Rotation: TSG2nSTR1.TSG2nTSF = 0 and TSG2nOPT0.TSG2nIDC = 0)

NOTE

TSG2nOPT0.TSG2nSOC = 0

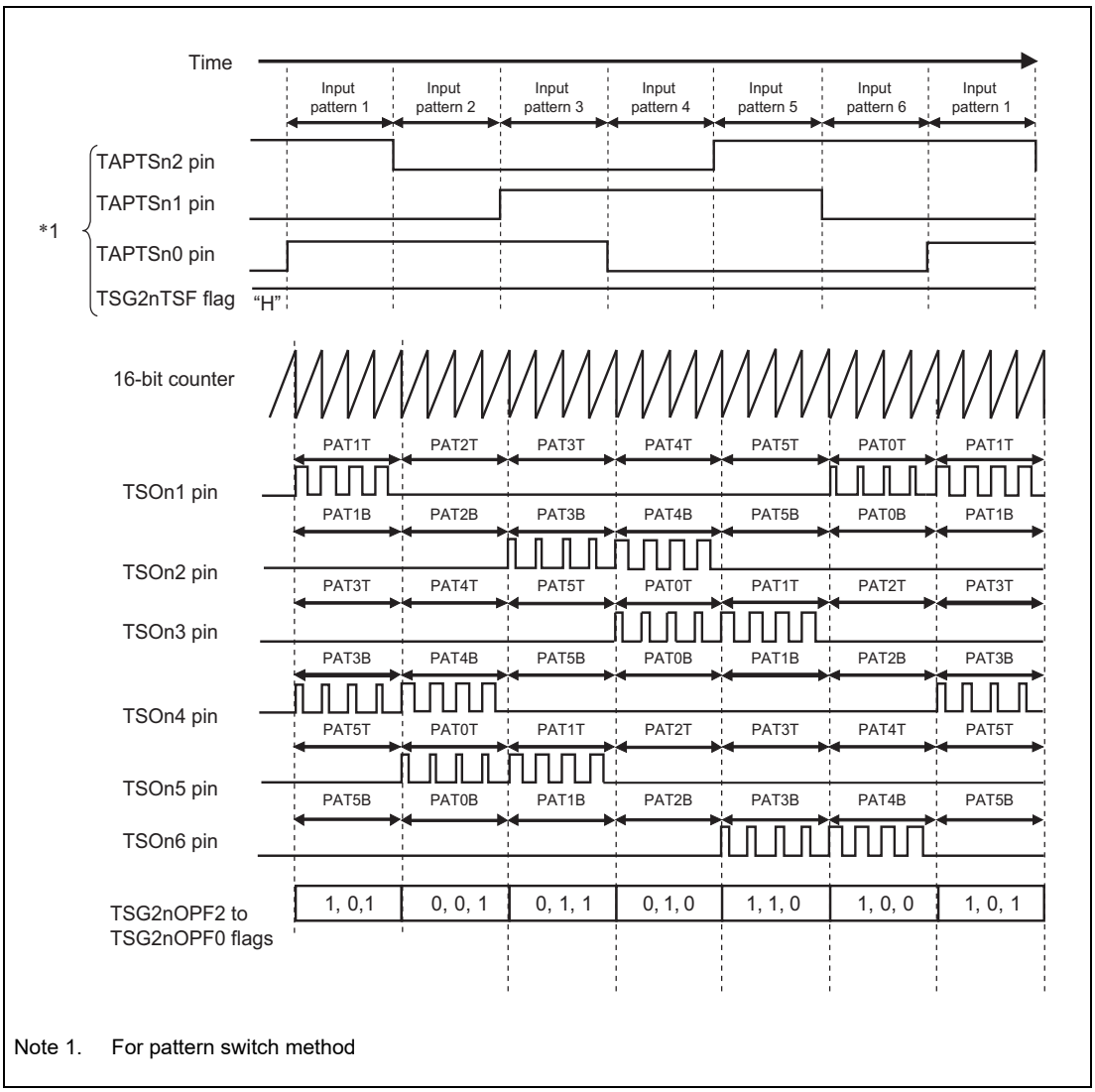


Note 1. For pattern switch method

Figure 23.75 Example of Operation in 120-DC Mode (Normal Rotation: TSG2nSTR1.TSG2nTSF = 0 and TSG2nOPT0.TSG2nIDC = 1)

NOTE

TSG2nOPT0.TSG2nSOC = 0

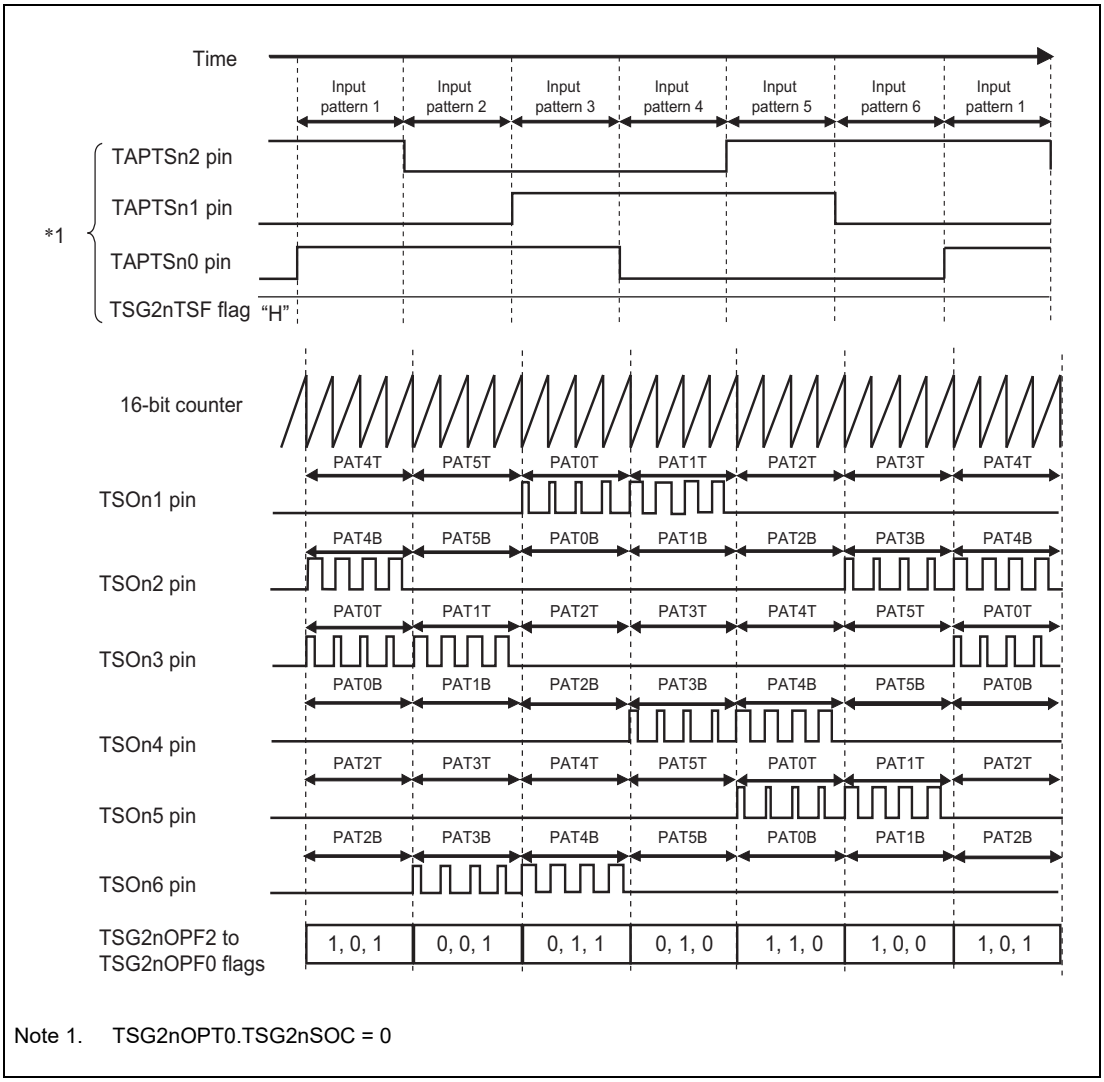


Note 1. For pattern switch method

Figure 23.76 Example of Operation in 120-DC Mode (Reverse Rotation: TSG2nSTR1.TSG2nTSF = 1 and TSG2nOPT0.TSG2nIDC = 0)

NOTE

TSG2nOPT0.TSG2nSOC = 0



Note 1. TSG2nOPT0.TSG2nSOC = 0

Figure 23.77 Example of Operation in 120-DC Mode (Reverse Rotation: TSG2nSTR1.TSG2nTSF = 1 and TSG2nOPT0.TSG2nIDC = 1)

NOTE

For pattern switch method

23.11.4.6 List of Output Patterns in 120-DC Mode

In 120-DC mode, the output pattern is determined according to the rotation direction and TSG2nOPT0.TSG2nIDC.

- Forward current direction (TSG2nIDC = 0)
  Forward pattern order direction (TSG2nTSF = 0 or TSG2nPSC = 0)

The order of pattern switching

	TSG2nOPT1.TSG2nSPC2 to TSG2nSPC0/TSG2nSTR1.TSG2nOPF2 to TSG2nOPF0							
Output Pin	**101**	**100**	**110**	**010**	**011**	**001**	**000**	**111**
TSOn1	PAT0T	PAT1T	PAT2T	PAT3T	PAT4T	PAT5T	Low	Low
TSOn2	PAT0B	PAT1B	PAT2B	PAT3B	PAT4B	PAT5B	Low	Low
TSOn3	PAT4T	PAT5T	PAT0T	PAT1T	PAT2T	PAT3T	Low	Low
TSOn4	PAT4B	PAT5B	PAT0B	PAT1B	PAT2B	PAT3B	Low	Low
TSOn5	PAT2T	PAT3T	PAT4T	PAT5T	PAT0T	PAT1T	Low	Low
TSOn6	PAT2B	PAT3B	PAT4B	PAT5B	PAT0B	PAT1B	Low	Low

- Reverse current direction (TSG2nIDC = 1)
  Forward pattern order direction (TSG2nTSF = 0 or TSG2nPSC = 0)

The order of pattern switching

	TSG2nOPT1.TSG2nSPC2 to TSG2nSPC0/TSG2nSTR1.TSG2nOPF2 to TSG2nOPF0							
Output Pin	**101**	**100**	**110**	**010**	**011**	**001**	**000**	**111**
TSOn1	PAT3T	PAT4T	PAT5T	PAT0T	PAT1T	PAT2T	Low	Low
TSOn2	PAT3B	PAT4B	PAT5B	PAT0B	PAT1B	PAT2B	Low	Low
TSOn3	PAT1T	PAT2T	PAT3T	PAT4T	PAT5T	PAT0T	Low	Low
TSOn4	PAT1B	PAT2B	PAT3B	PAT4B	PAT5B	PAT0B	Low	Low
TSOn5	PAT5T	PAT0T	PAT1T	PAT2T	PAT3T	PAT4T	Low	Low
TSOn6	PAT5B	PAT0B	PAT1B	PAT2B	PAT3B	PAT4B	Low	Low

NOTES

1. PAT0T to PAT5T: PWM output set by TSG2nCMP1W, TSG2nCMP5W, and TSG2nCMP9W
2. PAT0B to PAT5B: PWM output set by TSG2nCMP3W, TSG2nCMP7W, and TSG2nCMP11W

- Forward current direction (TSG2nIDC = 0)
  Reverse pattern order direction (TSG2nTSF = 1 or TSG2nPSC = 1)

The order of pattern switching

	TSG2nOPT1.TSG2nSPC2 to TSG2nSPC0/TSG2nSTR1.TSG2nOPF2 to TSG2nOPF0							
Output Pin	**101**	**100**	**110**	**010**	**011**	**001**	**000**	**111**
TSOn1	PAT1T	PAT0T	PAT5T	PAT4T	PAT3T	PAT2T	Low	Low
TSOn2	PAT1B	PAT0B	PAT5B	PAT4B	PAT3B	PAT2B	Low	Low
TSOn3	PAT3T	PAT2T	PAT1T	PAT0T	PAT5T	PAT4T	Low	Low
TSOn4	PAT3B	PAT2B	PAT1B	PAT0B	PAT5B	PAT4B	Low	Low
TSOn5	PAT5T	PAT4T	PAT3T	PAT2T	PAT1T	PAT0T	Low	Low
TSOn6	PAT5B	PAT4B	PAT3B	PAT2B	PAT1B	PAT0B	Low	Low

- Reverse current direction (TSG2nIDC = 1)
  Reverse pattern order direction (TSG2nTSF = 1 or TSG2nPSC = 1)

The order of pattern switching

	TSG2nOPT1.TSG2nSPC2 to TSG2nSPC0/TSG2nSTR1.TSG2nOPF2 to TSG2nOPF0							
Output Pin	**101**	**100**	**110**	**010**	**011**	**001**	**000**	**111**
TSOn1	PAT4T	PAT3T	PAT2T	PAT1T	PAT0T	PAT5T	Low	Low
TSOn2	PAT4B	PAT3B	PAT2B	PAT1B	PAT0B	PAT5B	Low	Low
TSOn3	PAT0T	PAT5T	PAT4T	PAT3T	PAT2T	PAT1T	Low	Low
TSOn4	PAT0B	PAT5B	PAT4B	PAT3B	PAT2B	PAT1B	Low	Low
TSOn5	PAT2T	PAT1T	PAT0T	PAT5T	PAT4T	PAT3T	Low	Low
TSOn6	PAT2B	PAT1B	PAT0B	PAT5B	PAT4B	PAT3B	Low	Low

NOTES

1. PAT0T to PAT5T : PWM output set by TSG2nCMP1W, TSG2nCMP5W, and TSG2nCMP9W
2. PAT0B to PAT5B : PWM output set by TSG2nCMP3W, TSG2nCMP7W, and TSG2nCMP11W

23.11.4.7 Operation Start Timing in 120-DC Mode

When pattern switch control is used, the pattern of the TAPTSn2 to TAPTSn0 pins can be detected but rotation direction (TSG2nSTR1.TSG2nTSF) cannot be determined. Therefore, set the rotation direction in TSG2nPSC when TSG2nTE is 0. The TSG2nPSC set value is loaded to TSG2nTSF, and the value can be used for the initial pattern setting.

- TSG2nOPT0.TSG2nSOC = 0, TSG2nPSC = 0, TSG2nIDC = 0

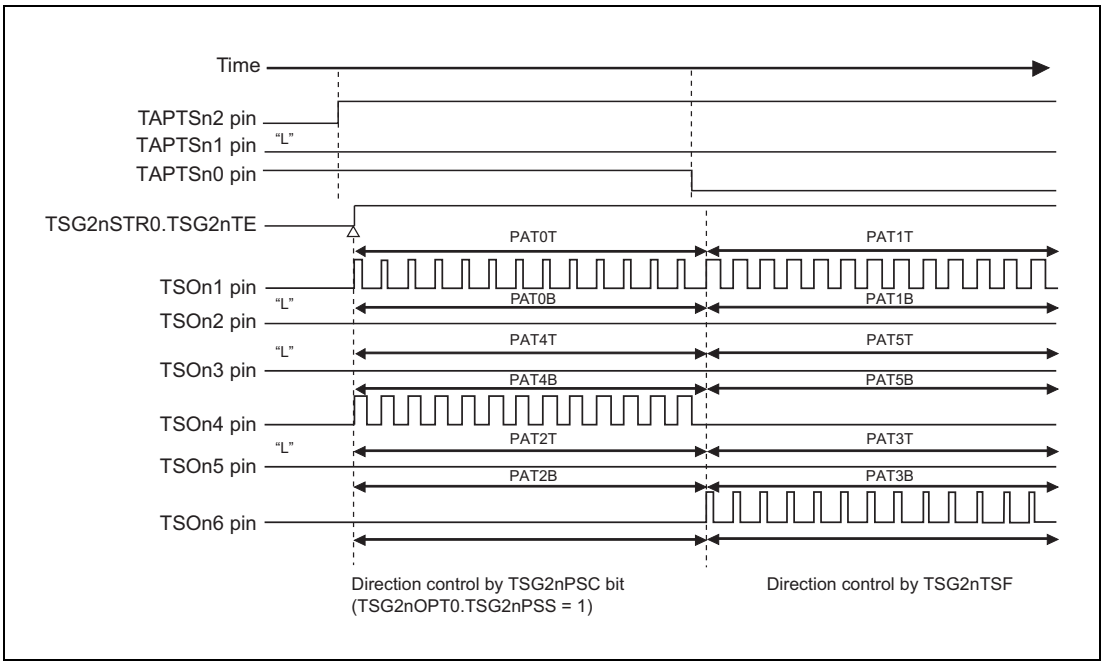


Figure 23.78 Control when Timer Output Starts in Normal Rotation (when Normal Pattern is Input)

- TSG2nOPT0.TSG2nSOC = 0, TSG2nPSC = 1, TSG2nIDC = 1

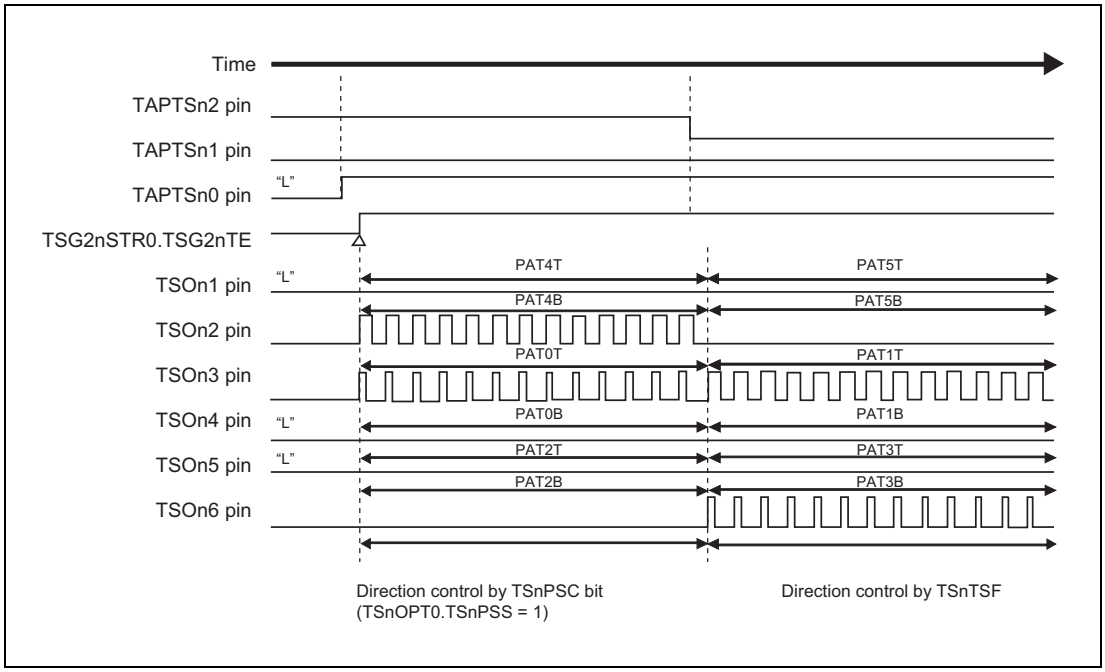


Figure 23.79 Control when Timer Output Starts in Reverse Rotation (when Normal Pattern is Input)

- TSG2nOPT0.TSG2nSOC = 0, TSG2nPSC = 0, TSG2nIDC = 0

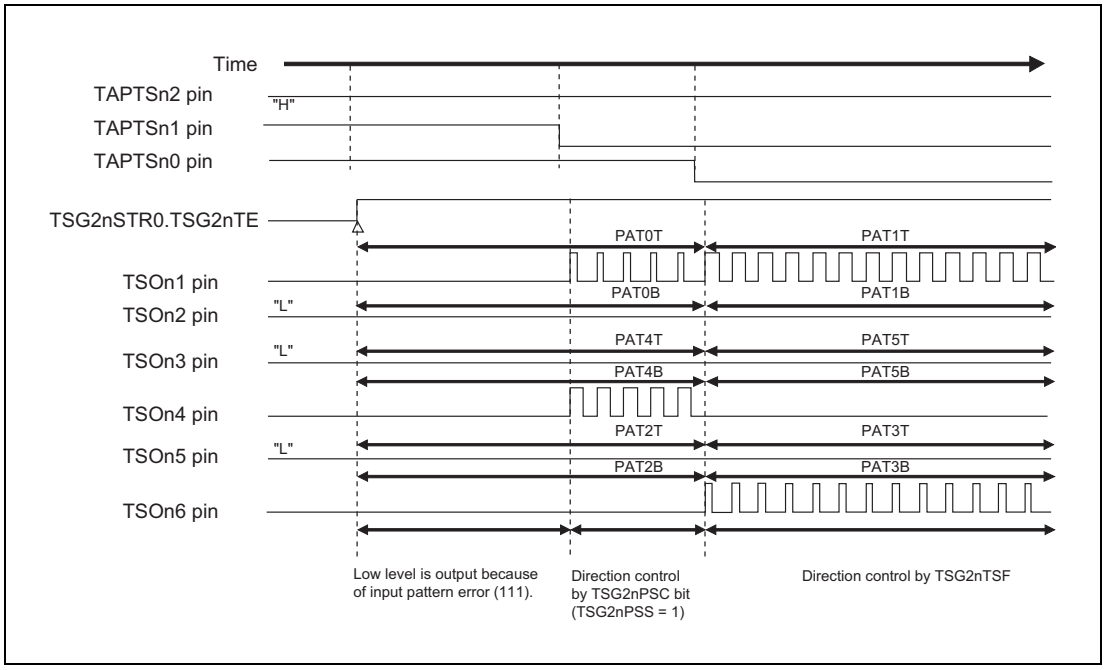


Figure 23.80 Control when Timer Output Starts in Normal Rotation (when Error Pattern is Input)

- TSG2nOPT0.TSG2nSOC = 0, TSG2nPSC = 1, TSG2nIDC = 1

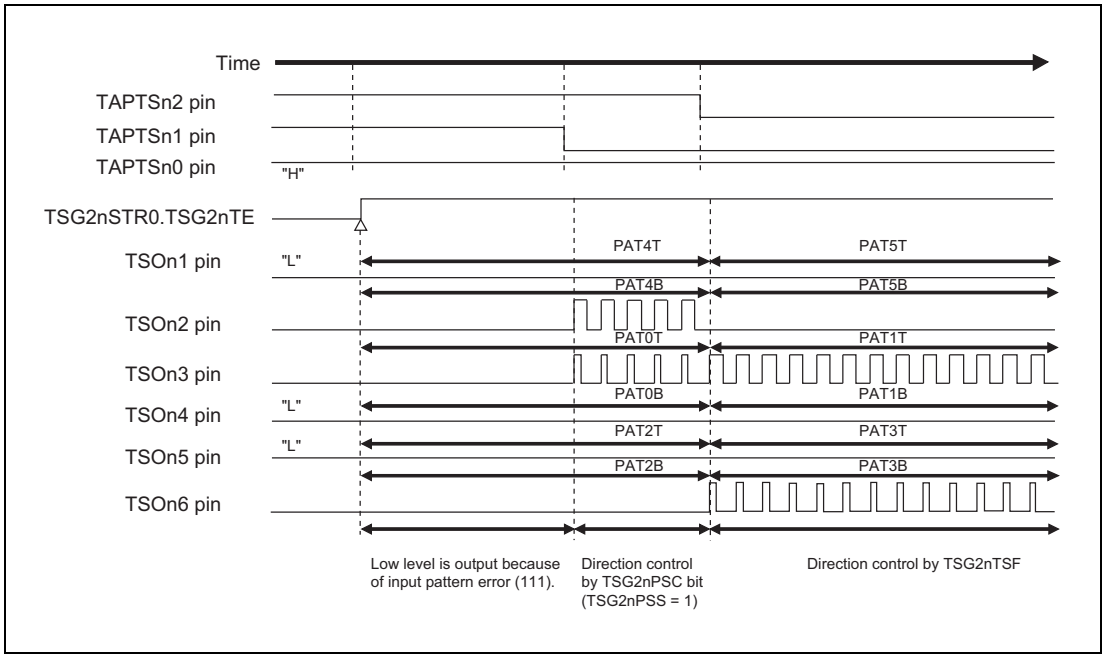


Figure 23.81 Control when Timer Output Starts in Reverse Rotation (when Error Pattern is Input)

23.11.4.8 Output Switch Timing in 120-DC Mode

As shown in **Figure 23.82** to **Figure 23.85**, in 120-DC mode, the external switch timing for output pattern (TAPTSn2 to TAPTSn0 pins) is input irrespective of the 16-bit counter operation. The output is switched to the new pattern by clearing the 16-bit counter using the pattern switch timing signal applied from outside.

In the pattern switch method, if a change in TAPTSn2 to TAPTSn0 pins occurs consecutively within one period, the value edge detected immediately before the period match timing is reflected to PWM output in the following periods.

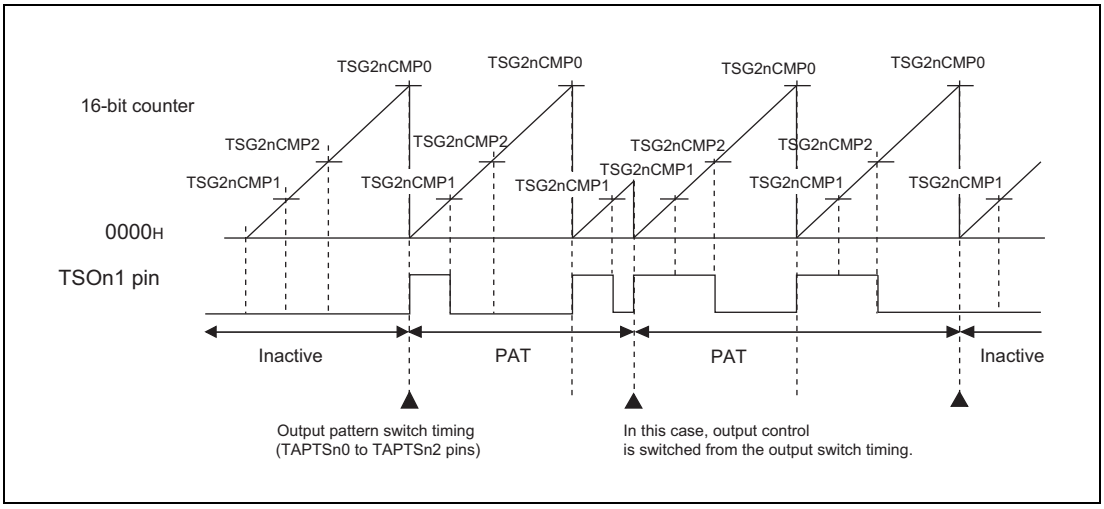
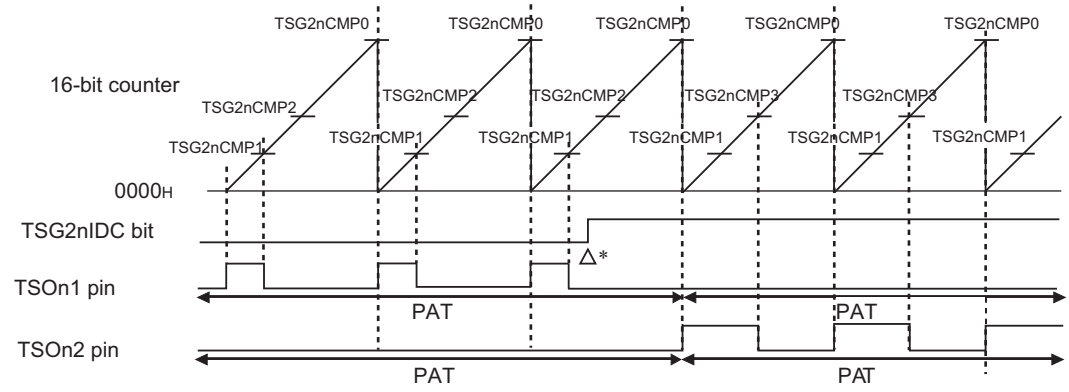


Figure 23.82 Output Switch Example (TAPTSn2 to TAPTSn0 Pins Trigger Input)

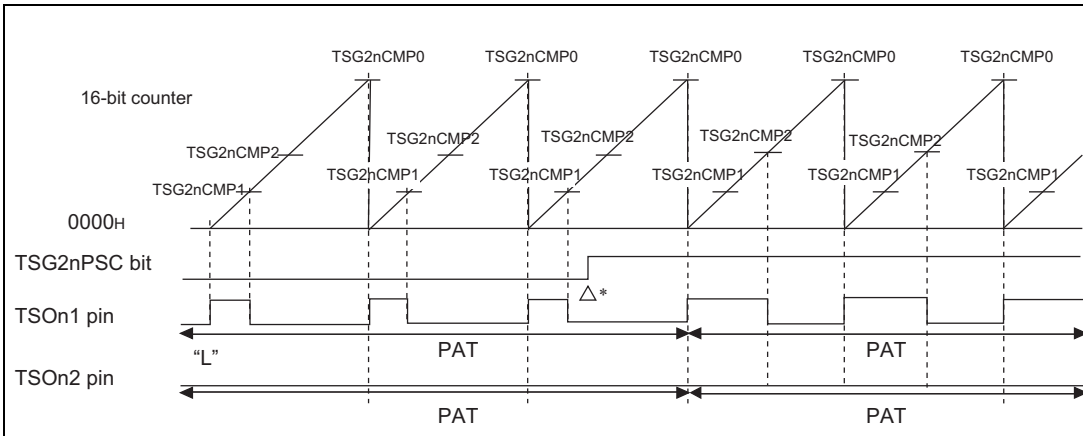


Note 1. When the pattern is switched here, output control is switched in the next period.

Figure 23.83 Output Switch Example (Switched by TSG2nOPT0.TSG2nIDC)

NOTE

If a change in the TAPTSn2 to TAPTSn0 pins occurs by the time the next period when output control is switched by the TSG2nIDC bit, the 16-bit counter is cleared and output control is switched.



Note 1. When the pattern is switched here, output control is switched from the next period.

Figure 23.84 Output Switch Example (Switched by TSG2nOPT0.TSG2nPSC)

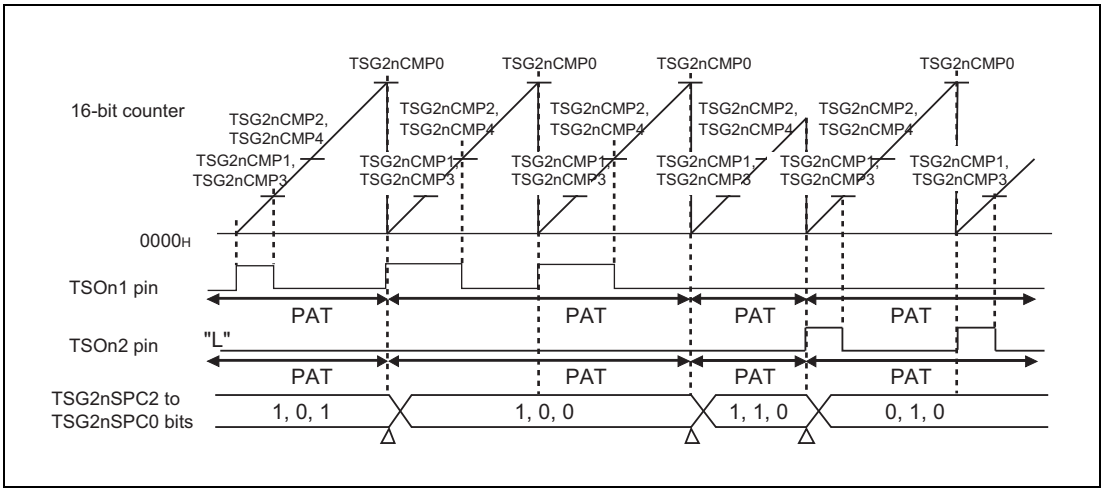


Figure 23.85 Output Switch Example (Switched by TSG2nOPT1.TSG2nSPC2 to TSG2nSPC0)

NOTE

Δ: Write access

23.11.4.9 Compare Register Rewrite Timing in 120-DC Mode

Example of operation when TSG2nCMP1 is reloaded (rewritten simultaneously) is shown below.

Figure 23.86 shows an output example when TSG2nCMP1 is rewritten. After TSG2nCMP1 is changed, data is not transferred to the TSG2nCMP1 buffer register (changed data is not valid) until the next reload timing; therefore, the specified output waveform can be obtained. However, do not write to TSG2nCMP1 again while the reload is suspended (period from when TSG2nCMP1 is changed to when simultaneous rewrite is executed). Be sure to read the reload request flag (TSG2nRSF) to confirm that the flag is 0, and write data to TSG2nCMP1.

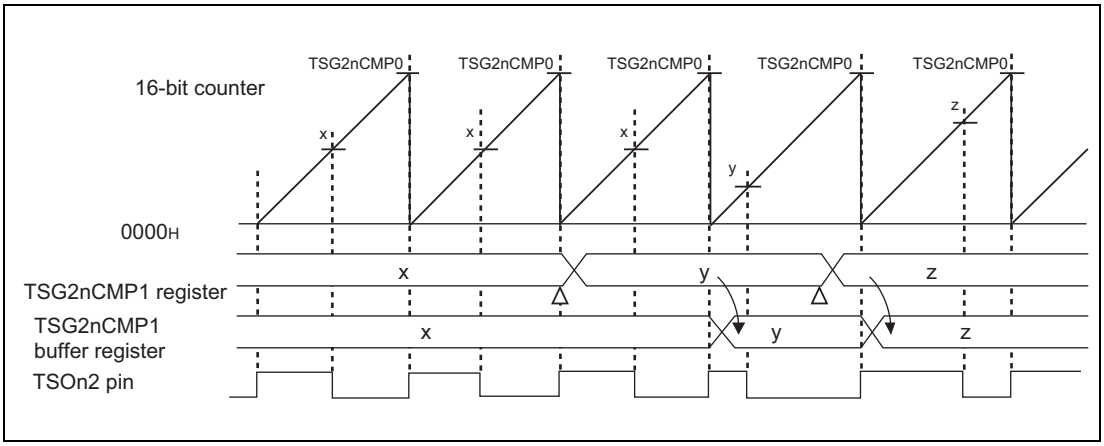


Figure 23.86 Output Example when TSG2nCMP1 is Rewritten

NOTE

Δ: Write access

23.11.4.10 Dead Time Control in 120-DC Mode

In 120-DC mode, the dead time is controlled on falling of each phase, and the dead time is added.

The dead time set in TSG2nDTC1 is inserted on falling of the positive phase, and the dead time set in TSG2nDTC0 is inserted on falling of the inverse phase.

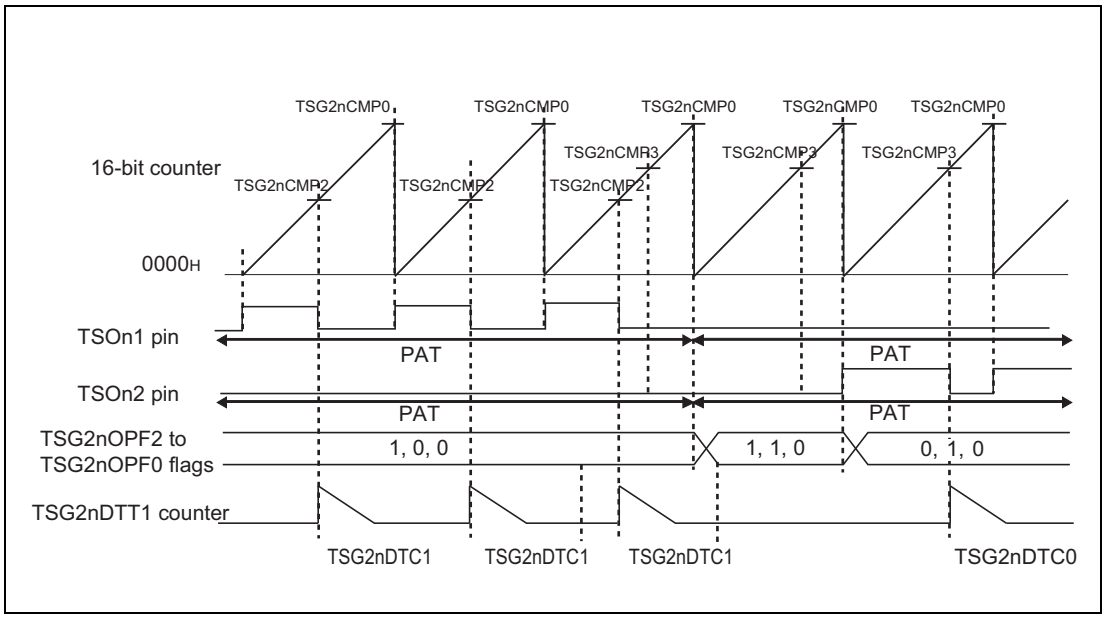


Figure 23.87 Output Switch Example

CAUTION

The dead time control method may affect the timer output. The timer output may not have the specified active level width due to the dead time control under the following conditions:

- When noise is generated on the input pattern in the pattern switch method
- When a change in the input pattern occurs earlier than the PWM period in the pattern switch method
- When switch method is changed
- When the current direction control bit (TSG2nOPT0.TSG2nIDC) is changed
- When the software output control function is used

23.11.4.11 Operation when Noise is Generated in TAPTSn2 to TAPTSn0 Pins in 120-DC Mode

Input to the TAPTSn2 to TAPTSn0 pins is assumed to be the hall sensor signals of the brushless DC motor. Depending on the system, a noise may be generated on the TAPTSn2 to TAPTSn0 pins. Operation when a noise is generated is described below.

For system product design, be sure to insert a noise filter circuit between the hall sensor and the TAPTSn2 to TAPTSn0 pins.

Figure 23.89 shows a case when a noise is generated on the TAPTSn2 to TAPTSn0 pins during operation with the pattern switch method used.

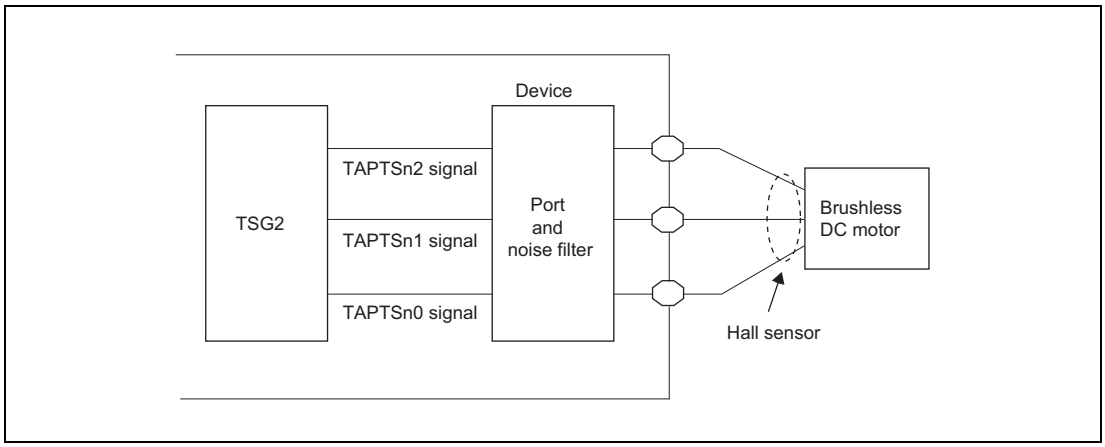


Figure 23.88 Example of Noise Filter Circuit Connection

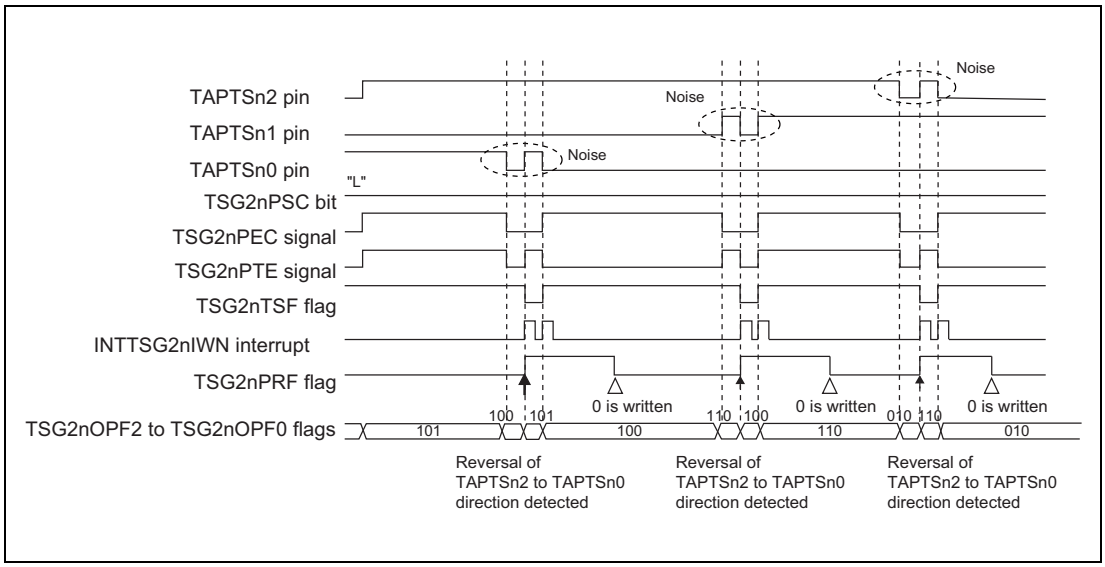


Figure 23.89 Example of Noise Generation at Level Change in TAPTSn2 to TAPTSn0 Pins (Pattern Switch Method)

(1) Change Timing of Input Pattern Change Detection Signal (TSG2nPTE)

- The TSG2nPTE signal toggles when the input pattern (TAPTSn2 to TAPTSn0 pins) changes.

CAUTION

Be sure to specify the rotation direction by TSG2nPSC (TSG2nPSS = 1 in TSG2nOPT0) in TSG2nOPT0.

(when TSG2nPSC = 0)

		TAPTSn2 to TAPTSn0 Pins after Change							
		000	**111**	**101**	**100**	**110**	**010**	**011**	**001**
Current TAPTSn2 to TAPTSn0 pins	000	—	—	—	—	—	—	—	—
	111	—	—	—	—	—	—	—	—
	101	—	—	—	Toggle	—	—	—	—
	100	—	—	—	—	Toggle	—	—	—
	110	—	—	—	—	—	Toggle	—	—
	010	—	—	—	—	—	—	Toggle	—
	011	—	—	—	—	—	—	—	Toggle
	001	—	—	Toggle	—	—	—	—	—

(when TSG2nPSC = 1)

		TAPTSn2 to TAPTSn0 Pins after Change							
		000	**111**	**101**	**100**	**110**	**010**	**011**	**001**
Current TAPTSn2 to TAPTSn0 pins	000	—	—	—	—	—	—	—	—
	111	—	—	—	—	—	—	—	—
	101	—	—	—	—	—	—	—	Toggle
	100	—	—	Toggle	—	—	—	—	—
	110	—	—	—	Toggle	—	—	—	—
	010	—	—	—	—	Toggle	—	—	—
	011	—	—	—	—	—	Toggle	—	—
	001	—	—	—	—	—	—	Toggle	—

(2) Change Timing of Three-Phase Encode Signal (TSG2nPEC)

- The TSG2nPEC signal toggles when input pattern (TAPTSn2 to TAPTSn0 pins) changes.

		TAPTSn2 to TAPTSn0 Pins after Change							
		000	111	101	100	110	010	011	001
Current TAPTSn2 to TAPTSn0 Pins	000	—	—	—	—	—	—	—	—
	111	—	—	—	—	—	—	—	—
	101	—	—	—	Toggle	—	—	—	Toggle
	100	—	—	Toggle	—	Toggle	—	—	—
	110	—	—	—	Toggle	—	Toggle	—	—
	010	—	—	—	—	Toggle	—	Toggle	—
	011	—	—	—	—	—	Toggle	—	Toggle
	001	—	—	Toggle	—	—	-	Toggle	—

(3) Change Timing of TSOn1 to TSOn6 Pins

- When the pattern switch method is used, the output pattern changes when the input signal of the TAPTSn2 to TAPTSn0 pins changes. The output is also switched when two or more pins change simultaneously.

CAUTION

When the input pattern changes to 000 or 111, the TSOn1 to TSOn6 pins are driven low.

(4) Change Timing of TSG2nTSF Flag

- The TSG2nTSF flag toggles when the input pattern (TAPTSn2 to TAPTSn0 pins) changes.

		TAPTSn2 to TAPTSn0 Pins after Change							
		000	111	101	100	110	010	011	001
Current TAPTSn2 to TAPTSn0 Pins	000	—	—	—	—	—	—	—	—
	111	—	—	—	—	—	—	—	—
	101	—	—	—	0	—	—	—	1
	100	—	—	1	—	0	—	—	—
	110	—	—	—	1	—	0	—	—
	010	—	—	—	—	1	—	0	—
	011	—	—	—	—	—	1	—	0
	001	—	—	0	—	—	—	1	—

(5) Set Timing of TSG2nNDF Flag

- The TSG2nNTF flag is set when two or more pins of the TAPTSn2 to TAPTSn0 pins change simultaneously, and cleared when 1 is written to the TSG2nNDR bit. The TSG2nNDF flag is valid when 1 is set to the TSG2nNDC bit.

(6) Set Timing of TSG2nPRF Flag

- The TSG2nPRF flag is set when the TSG2nTSF flag changes, and cleared when 1 is written to the TSG2nPRR bit. The TSG2nPRF flag is valid when 1 is set to the TSG2nPRC bit.

(7) Set Timing of TSG2nPEF Flag

- The TSG2nPRF flag is set when 000 or 111 is input to the TAPTSn2 to TAPTSn0 pins, and cleared when 1 is written to the TSG2nPER bit. The TSG2nPEF flag is valid when the TSG2nPEC bit is set to 1.

23.11.4.12 Basic Control Flow in 120-DC Mode

In 120-DC mode, there are eight control states as listed in **Table 23.73**.

When TSG2nOPT0.TSG2nSTE = 1, the pattern switch method is used for 120-DC control.

This is defined as fixed phase control. The fixed phase control should be performed considering the factors such as delay from the hall sensor and delay from sensor level detection to timer output. However, acceleration or deceleration can be performed simply by changing the PWM duty.

Table 23.73 Timer Control Status

Status	TSG2nTSF in TSG2nSTR1	TSG2nIDC in TSG2nOPT0	Control
A	0	0	Normal rotation, acceleration, and fixed phase
D	0	1	Normal rotation, deceleration, and fixed phase
E	1	1	Reverse rotation, acceleration, and fixed phase
H	1	0	Reverse rotation, deceleration, and fixed phase

Generally, the state, when the motor rotation stops, is assumed to be a state of the start and the control begins. First the fixed phase control is used to rotate the motor from the stopped state.

State transition is shown in **Figure 23.90**.

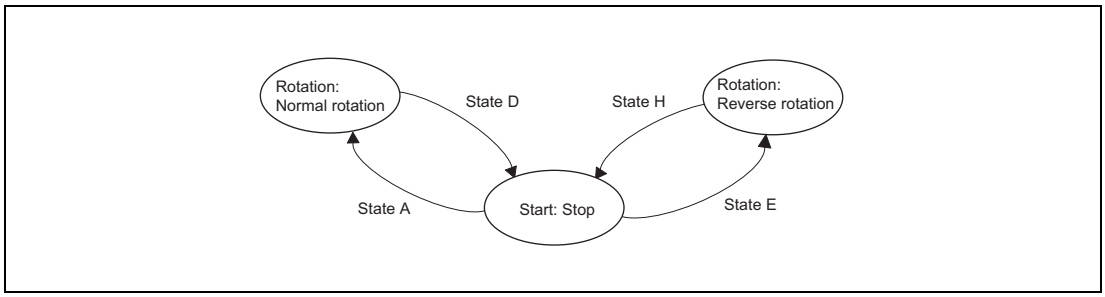


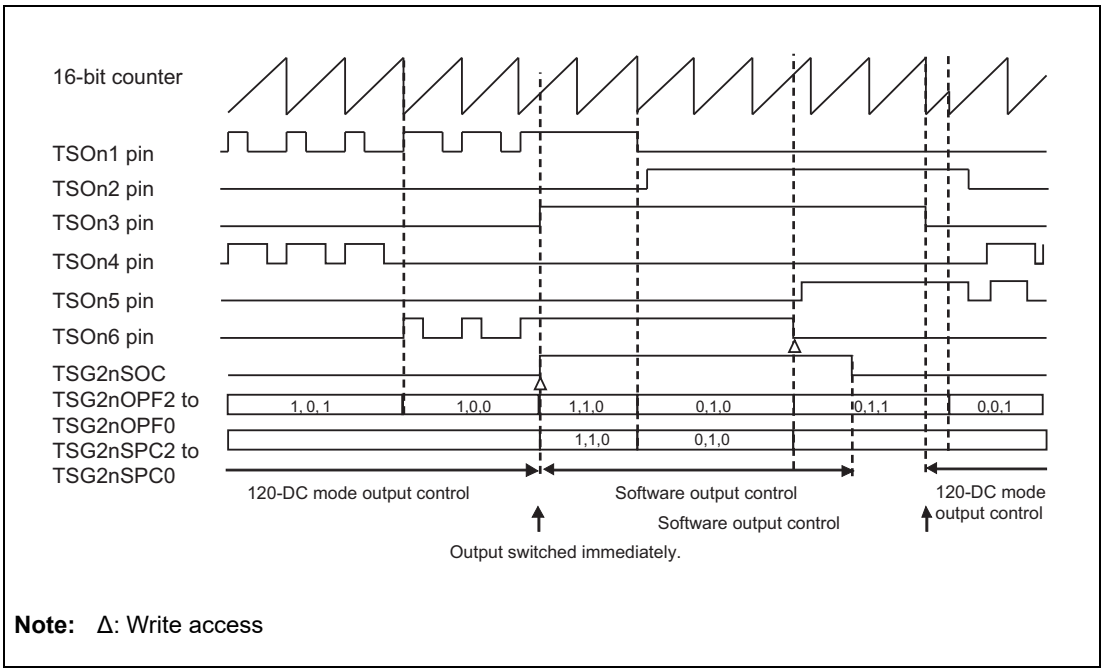
Figure 23.90 State Transition Diagram

23.11.4.13 Software Output Control Function in 120-DC Mode

TSG2nOPT0.TSG2nSOC and TSG2nIDC, and TSG2nOPT1.TSG2nSPC2 to TSG2nSPC0 are used in 120-DC mode for timer output control by software.

As shown in **Figure 23.91**, the output control is switched immediately when TSG2nSOC is set to 1. If the dead time is set, the period of the dead time period is guaranteed. After that, to switch the output control from software output control to 120-DC control, set TSG2nSOC to 0. At this timing, output control is retained. When the reload timing is generated, output control is switched to 120-DC mode.

For details on software output control function, see **Section 23.11.5, Software Output Control Function**.



Note: Δ: Write access

Figure 23.91 Example of Switching from 120-DC Mode to Software Output Control Function

(1) Procedure for Software Output Control

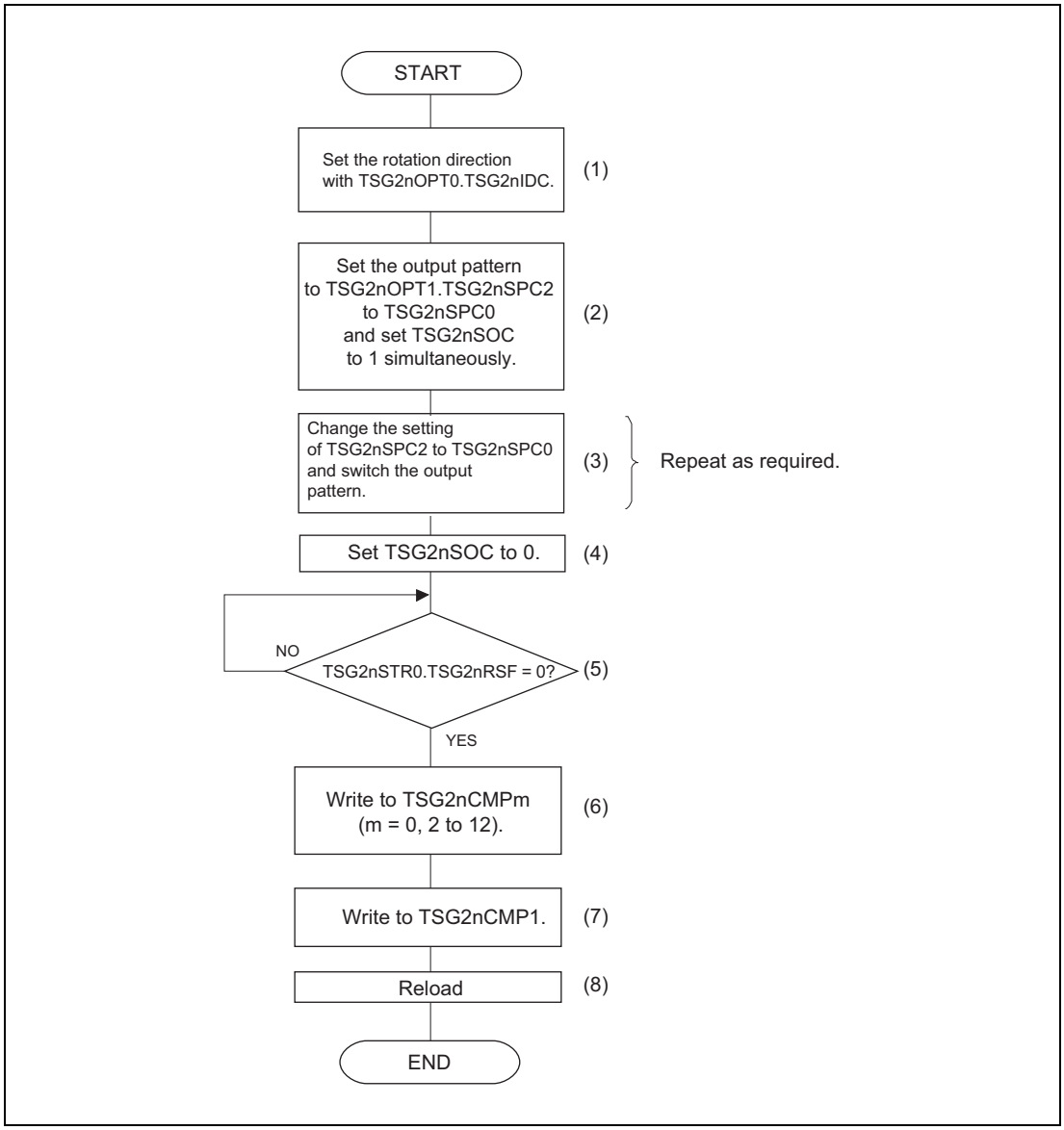


Figure 23.92 Process Flow of Software Output Control

The procedure for software output control is described below.

(1) Set TSG2nIDC to determine the rotation direction. The phase of the timer output with TSG2nIDC = 0 is different by 180 degrees from that with TSG2nIDC = 1. The timer output does not change if only this bit is rewritten using the software output control function. However, if the period match occurs before step (2), the output pattern of 120-DC control changes. So, schedule so as to prevent the period match from occurring before step (2).

(2) Set the output pattern to TSG2nSPC2 to TSG2nSPC0. To enable software output control, set TSG2nSOC to 1 simultaneously.

(3) Change the output pattern setting of TSG2nSPC2 to TSG2nSPC0 to change the timer output. The registers that can be changed during software control are: TSG2nTRG1.TSG2nTT, TSG2nCTL4 to TSG2nCTL6, TSG2nOPT0, TSG2nOPT1, TSG2nCMP0 to TSG2nCMP12, TSG2nDTC0, and TSG2nDTC1.

(4) Confirm that the reload request flag (TSG2nRSF) = 0. If TSG2nRFS = 1, do not proceed to the following step until TSG2nRSF = 0.

(5) By setting TSG2nSOC = 0 the software control starts to be released (it is not released here yet).

(6) After releasing the software output control, set the necessary compare registers. Proceed to the following step when the register setting is not required. Here, change the registers with the reload function.

(7) Write to TSG2nCMP1 to start reloading.

(8) Reload is executed and software output is released.

CAUTION

Execute reload after executing steps (4) to (7). When reload cannot be executed, the software output cannot be released.

23.11.5 Software Output Control Function

Software output control function can be used in HT-PWM mode, SP-PWM mode, and 120-DC mode. This function can switch six output patterns for the TSOn1 to TSOn6 pins using TSG2nOPT0.TSG2nSOC, TSG2nIDC, and TSG2nOPT1.TSG2nSPC2 to TSG2nSPC0.

When TSG2nSOC is switched from 0 to 1, the output control method of the TSOn1 to TSOn6 pins is switched to the software output control immediately. On the contrast, when TSG2nSOC is switched from 1 to 0, the software output control is released at the reload timing.

Table 23.74 Register Description on Software Output Control Function

Register	Setting
TSG2nOPT0.TSG2nSOC	TSG2nSOC = 1
TSG2nOPT0.TSG2nSTE	TSG2nSTE = 0
TSG2nOPT1.TSG2nSPC2 to TSG2nSPC0	Set output patterns listed in **Table 23.75** and **Table 23.76**.
TSG2nOPT0.TSG2nIDC	Set output pattern (rotation direction).

Table 23.75 Output Pattern of Software Output Control (TSG2nOPT0.TSG2nIDC = 0)

TSG2nOPT0.TSG2nSOC = 1, TSG2nSTE = 0, TSG2nIDC = 0

	TSG2nSTR1.TSG2nOPF2 to TSG2nOPF0							
Output Pin	101	100	110	010	011	001	000	111
TSOn1	ACT	ACT	ACT	INACT	INACT	INACT	INACT	ACT
TSOn2	INACT	INACT	INACT	ACT	ACT	ACT	ACT	INACT
TSOn3	INACT	INACT	ACT	ACT	ACT	INACT	INACT	ACT
TSOn4	ACT	ACT	INACT	INACT	INACT	ACT	ACT	INACT
TSOn5	ACT	INACT	INACT	INACT	ACT	ACT	INACT	ACT
TSOn6	INACT	ACT	ACT	ACT	INACT	INACT	ACT	INACT

Note: ACT: Active level is output.
INACT: Inactive level is output.

Table 23.76 Output Pattern of Software Output Control (TSG2nOPT0.TSG2nIDC = 1)

TSG2nOPT0.TSG2nSOC = 1, TSG2nSTE = 0, TSG2nIDC = 1

	TSG2nSTR1.TSG2nOPF2 to TSG2nOPF0							
Output Pin	101	100	110	010	011	001	000	111
TSOn1	INACT	INACT	INACT	ACT	ACT	ACT	ACT	INACT
TSOn2	ACT	ACT	ACT	INACT	INACT	INACT	INACT	ACT
TSOn3	ACT	ACT	INACT	INACT	INACT	ACT	ACT	INACT
TSOn4	INACT	INACT	ACT	ACT	ACT	INACT	INACT	ACT
TSOn5	INACT	ACT	ACT	ACT	INACT	INACT	ACT	INACT
TSOn6	ACT	INACT	INACT	INACT	ACT	ACT	INACT	ACT

Note: ACT: Active level is output.
INACT: Inactive level is output.

Section 24 Timer Option (TAPA)

24.1 Overview

The timer option (TAPA) is for use with the motor control timer (TSG2) to realize a motor system.

- Asynchronous Hi-Z control to deal with pin input

24.1.1 Basic functions

Channel

This product has the following number of channels of timer option modules.

Table 24.1 Channels of Timer Option Modules

Timer Option	
Number of channels	2
Name	TAPAn (n = 2, 3)

Meaning of n

Throughout this section, the individual channels of the timer option function are identified by the index “n”. For example, TAPAnFLG indicates the TAPAn flag registers (n = 2, 3).

Since TAPA0 and 1 are not used in this product, only TAPA2 and TAPA3 are provided.

Register addresses

All TAPAn register addresses are given as offsets from the given base address, <TAPAn_base>. Each of the TAPAn base addresses is listed in the following table.

Table 24.2 Register Base Address

TAPAn	<TAPAn_base> Address
TAPA2	FFE9 2000_H
TAPA3	FFE9 3000_H

24.2 Registers

TAPAn (n = 2 and 3) is controlled and operated by means of the following registers.

24.2.1 Registers Overview

The list of TAPAn (n = 2 and 3) registers and the memory addresses are shown below.

The base address is FFE9 2000_H for TAPA2 and FFE9 3000_H for TAPA3.

For the actual addresses, the offset values indicated in the following table are added to the base addresses.

Register Name	Function	R/W	Reset Value	Access Unit			Offset Address
				8	16	32	
TAPAnCTL0	TAPAn control register 0	R/W	0000_H	—	√	—	0020_H
TAPAnFLG	TAPAn flag register	R	0000_H	—	√	—	0000_H
TAPAnACWE	TAPAn asynchronous control write enable register	R/W	00_H	√	—	—	0004_H
TAPAnACTS	TAPAn asynchronous control start trigger register	W	00_H	√	—	—	0008_H
TAPAnACTT	TAPAn asynchronous control stop trigger register	W	00_H	√	—	—	$000C_H$
TAPAnOPHS	TAPAn Hi-Z start trigger register	W	00_H	√	—	—	0014_H
TAPAnOPHT	TAPAn Hi-Z stop trigger register	W	00_H	√	—	—	0018_H

24.2.2 TAPAnCTL0 — TAPAn Control Register 0

Control register 0 is used to control Hi-Z.

A value in this register can only be rewritten when TAPAnFLG.TAPAnACE = 0.

Access: This register can be read/written in 16-bit units.

Address: <TAPAn_base> + 20_H

Value after reset: 0000_H
This register is initialized by a reset from any source.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	TAPAn DCM	TAPAn DCN	TAPAn DCP	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R	R

Table 24.3 TAPAnCTL0 Register Contents

Bit Position	Bit Name	Function
15 to 5	Reserved	When read, the value after reset is read. When writing, write the value after reset.
4	TAPAnDCM	Clearing Condition Specification This control bit specifies the condition for clearing of the Hi-Z control outputs. 0: Manipulation of TAPAnOPHT0 is enabled regardless of the TAPATHASIN signal input level. 1: Manipulation of TAPAnOPHT0 is disabled when the TAPATHASIN input signal is at the active level. Manipulation of TAPAnOPHT0 is enabled when the TAPATHASIN signal input is inactive.
3, 2	TAPAnDCN, TAPAnDCP	Hi-Z Input Edge Selection These control bits specify the effective edge of TAPAnTHASIN.
1, 0	Reserved	When read, the value after reset is read. When writing, write the value after reset.

TAPAn DCN	TAPAn DCP	Description
0	0	Does not detect effective edges
0	1	Detects a rising edge as the effective edge (active level = high)
1	0	Detects a failing edge as the effective edge (active level = low)
1	1	Setting is prohibited.

24.2.3 TAPAnFLG — TAPAn Flag Register

This flag register is used to control Hi-Z.

Access: This register can be read/written in 16-bit units.

Address: <TAPAn_base> + 00_H

Value after reset: 0000_H
This register is initialized by a reset from any source.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	TAPAnHOF[2:0]			—	—	—	—	—	—	—	TAPAn ACE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 24.4 TAPAnFLG Register Contents

Bit Position	Bit Name	Function
15 to 11	Reserved	When read, the value after reset is read. When writing, write the value after reset.
10 to 8	TAPAnHOF[2:0]	HZOUTm Output Monitor (m = 0 to 2) These bits monitor the TAPATHZOUTmz output. 0: The TAPATHZOUTmz output is at the high level. 1: The TAPATHZOUTmz output is at the low level.
7 to 1	Reserved	When read, the value after reset is read. When writing, write the value after reset.
0	TAPAnACE	Asynchronous Hi-Z Control Enable This bit indicates the state of asynchronous Hi-Z control (TAPATHASIN). 0: Asynchronous Hi-Z control is stopped. 1: Asynchronous Hi-Z control is enabled. The conditions for setting and clearing this bit are as follows. Clearing condition: Writing 1 to TAPAnACTT while TAPAnACWE = 1. Setting condition: Writing 1 to TAPAnACTS while TAPAnACWE = 1.

24.2.4 TAPAnACWE — TAPAn Asynchronous Control Write Enable Register

This register enables writing for asynchronous Hi-Z control.

Access: This register can be read/written in 8-bit units.

Address: <TAPAn_base> + 04_H

Value after reset: 00_H
This register is initialized by a reset from any source.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TAPAn ACWE
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 24.5 TAPAnACWE Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When read, the value after reset is read. When writing, write the value after reset.
0	TAPAnACWE	Asynchronous Control Write Enable This is a write-enable bit for asynchronous Hi-Z control. After 1 has been written to this bit, it is automatically cleared to 0 by writing 1 to TAPA0ACTS and TAPA0ACTT. 0: Disables writing to TAPAnACTS and TAPAnACTT. 1: Enables writing to TAPAnACTS and TAPAnACTT.

24.2.5 TAPAnACTS — TAPAn Asynchronous Control Start Trigger Register

This register enables the start trigger for asynchronous Hi-Z control.

Access: This register can be read/written in 8-bit units. This register is always read as 00_H.

Address: <TAPAn_base> + 08_H

Value after reset: 00_H
This register is initialized by a reset from any source.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TAPAn ACTS
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 24.6 TAPAnACTS Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When read, the value after reset is read. When writing, write the value after reset.
0	TAPAnACTS	Asynchronous Control Start Trigger This bit enables the start trigger for asynchronous Hi-Z control. The setting of this bit is only valid when TAPAnACWE = 1. 0: Writing 0 to this bit is ignored (no function). 1: Enables asynchronous Hi-Z control if TAPAnACWE = 1.

24.2.6 TAPAnACTT — TAPAn Asynchronous Control Stop Trigger Register

This register enables the stop trigger for asynchronous Hi-Z control.

Access: This register can be read/written in 8-bit units. This register is always read as 00_H.

Address: <TAPAn_base> + $0C_H$

Value after reset: 00_H
This register is initialized by a reset from any source.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TAPAn ACTT
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 24.7 TAPAnACTT Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When read, the value after reset is read. When writing, write the value after reset.
0	TAPAnACTT	Asynchronous Control Stop Trigger This bit enables the stop trigger for asynchronous Hi-Z control. The setting of this bit is only valid when TAPAnACWE = 1. 0: Writing 0 to this bit is ignored (no function). 1: Stops asynchronous Hi-Z control if TAPAnACWE = 1.

24.2.7 TAPAnOPHS — TAPAn Hi-Z Start Trigger Register

This register sets the start trigger for a Hi-Z control signal (TAPATHZOUT0Z).

Access: This register can be read/written in 8-bit units. This register is always read as 00_H.

Address: <TAPAn_base> + 14_H

Value after reset: 00_H
This register is initialized by a reset from any source.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TAPAn OPHS0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 24.8 TAPAnOPHS Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When read, the value after reset is read. When writing, write the value after reset.
0	TAPAnOPHS0	Hi-Z Control Signal Start Trigger 0 This bit sets the start trigger for a Hi-Z control signal. 0: The read value is always 0 and writing 0 to this bit is ignored. 1: Sets the Hi-Z control signal (TAPATHZOUT0Z) to the low level.

24.2.8 TAPAnOPHT — TAPAn Hi-Z Stop Trigger Register

This register sets the stop trigger for a Hi-Z control signal (TAPATHZOUT0Z).

Access: This register can be read/written in 8-bit units. This register is always read as 00_H.

Address: <TAPAn_base> + 18_H

Value after reset: 00_H
This register is initialized by a reset from any source.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TAPAn OPHT0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 24.9 TAPAnOPHT Register Contents

Bit Position	Bit Name	Function
7 to 1	Reserved	When read, the value after reset is read. When writing, write the value after reset.
0	TAPAnOPHT0	Hi-Z Control Signal Stop Trigger 0 This bit sets the stop trigger for a Hi-Z control signal. 0: The read value is always 0 and writing 0 to this bit is ignored. 1: Sets a Hi-Z control signal (TAPATHZOUT0Z) to the high level.

24.3 Basic Functions

24.3.1 Hi-Z Control Function

24.3.1.1 Purpose of Hi-Z Control Function

Abnormal operation in timer motor-control under CPU control leads to rotation of the externally connected motor also becoming abnormal. In such a case, this function forcibly sets the motor control output to the Hi-Z state, independently of control by the CPU.

24.3.1.2 Overview of Hi-Z Control Function

The following method is available for controlling Hi-Z.

- Asynchronous input Hi-Z control for pin inputs
  - Controls the Hi-Z control output signals TAPATHZOUT0Z (phase U) asynchronously.

24.3.1.3 Hi-Z Control and Its Operation

Function	Operation
Asynchronous Hi-Z control corresponding to pin input	This function detects pin inputs that are asynchronous and forcibly stops TSOn1 to 6 pin output from the corresponding timer module (TSG2) in response. Device outputs become Hi-Z while TAPATHASIN is active and until software sends a stop request.

24.3.2 Asynchronous Hi-Z Control for pin Inputs

24.3.2.1 Basic Operation in Hi-Z Control Asynchronous Inputs

Examples of system configuration are described below.

Hi-Z Control when TAPAnCTL0.TAPAnDCM = 0, TAPAnDCP = 1, and TAPAnDCN = 0

TAPATHZOUT0Z goes to the low level on detection of an effective edge of the asynchronous input (TAPATHASIN).

Output is forcibly stopped (by port control for Hi-Z output) as long as the TAPATHZOUT0Z output is at the low level.

TAPATHZOUT0Z goes to the high level in response to writing 1 to Hi-Z stop trigger 0 (TAPAnOPHT0), regardless of the level of TAPATHASIN.

Hi-Z Control when TAPAnCTL0.TAPAnDCM = 1, TAPAnDCP = 1, TAPAnDCN = 0

TAPATHZOUT0Z goes to the low level in response to detection of an effective edge of the asynchronous input signal (TAPATHASIN).

Output is forcibly stopped (by port control for Hi-Z output) as long as the TAPATHZOUT0Z output is at the low level.

Writing of 1 to Hi-Z stop trigger 0 (TAPAnOPHT0) is ignored as long as the asynchronous input signal (TAPATHASIN) is at the active level (high because TAPAnDCP = 1).

After the asynchronous input signal (TAPATHASIN) is switched to the inactive level (low because TAPAnDCP = 1), TAPATHZOUT0 goes to the high level when 1 is written to Hi-Z stop trigger 0 (TAPAnOPHT0).

24.3.2.2 Software Operations for Asynchronous Input Hi-Z Control

This module allows software control of the output of Hi-Z control signals.

Hi-Z start trigger 0 (TAPAnOPHS0) and Hi-Z stop trigger 0 (TAPAnOPHT0) are used to control TAPATHZOUT0Z.

Operation of the Hi-Z Start Trigger (TAPAnOPHS)

TAPAnDCM	Operation
0/1	Writing 1 to the TAPAnOPHS0 bit places the TAPATHZOUT0Z signal at the low level.

Operation of the Hi-Z Stop Trigger (TAPAnOPHT) during Hi-Z Control in Response to Asynchronous Input

The Hi-Z stop trigger operates as follows.

TAPAnDCM	Operation
0	Writing 1 to the TAPAnOPHT0 bit places the TAPATHZOUT0Z signal at the high level.
1	If TAPATHASIN is at the inactive level, writing 1 to the TAPAnOPHT0 bit places the TAPATHZOUT0Z signal at the high level. If TAPATHASIN is at the active level, writing of 1 to the TAPAnOPHT0 bit is ignored.

24.3.2.3 Operating Procedure Example for Hi-Z Control in Response to Asynchronous Input

An example of the operating procedure for Hi-Z control in response to asynchronous input is as follows (the table only covers settings for the timer option module because this operation does not depend on timer operations).

		Operation	State of TAPA
Restart operation	Initial settings	Setting in the TAPAnCTL0 register. Set TAPAnDCP and TAPAnDCN (input edge selection). Set TAPAnDCM (clearing mode selection).	Hi-Z control in response to asynchronous input is stopped (TAPAnFLG.TAPAnACE = 0).
	Starting operation	Setting in the TAPAnACWE register: Set the TAPAnACWE bit to 1. Setting in the TAPAnACTS register: Set the TAPAnACTS bit to 1.	Writing to the TAPAnACTS bit is enabled. TAPAnFLG.TAPAnACE = 1, enabling Hi-Z control in response to asynchronous input.
	During operation	To start Hi-Z control of an output from the timer: – Control is by the TAPAnOPHS0 bit of TAPA – Control is by the Hi-Z input signal (TAPATHASIN) for TAPA To stop Hi-Z control of output from the timer: – Control is by the TAPAnOPHT0 bit of TAPA (if TAPAnDCM = 0) – TAPAnOPHT0 is used if the Hi-Z input signal for TAPA (TAPATHASIN) is at the inactive level (if TAPAnDCM = 1) The state of TAPA operations can be read from the TAPAnFLG register at all times.	On detection of input of the starting edge of the Hi-Z input signal (TAPATHASIN) or setting of the start trigger bit (TAPAnOPHS0 = 1), the Hi-Z controller switches the TAPATHZOUT0Z pin to low-level output. In accord with the operating mode settings in TAPAnDCM, the Hi-Z controller switches the TAPATHZOUT0Z pins to high-level outputs in response to setting of the stop trigger bit (TAPAnOPHT0 = 1).
	Stopping operation	Setting in the TAPAnACWE register: Set the TAPAnACWE bit to 1. Setting in the TAPAnACTT register. Set the TAPAnACTT bit to 1.	Writing to the TAPAnACTT bit is enabled. TAPAnFLG.ACE = 0, stopping asynchronous Hi-Z control.

Section 25 Peripheral Interconnection (PIC)

25.1 Timer Synchronization and Port Hi-Z Function

25.1.1 Overview

The peripheral interconnection (PIC1) handles synchronous operation using multiple timers and Hi-Z control of ports by working with TAPA.

Table 25.1 describes the overview of the PIC1 specification.

Table 25.1 PIC1 Specification Overview

Item	Description
Timer synchronization	Realizes simultaneous start of any combination of timers (TSG2n and OSTMn).
Port Hi-Z function	Realizes Hi-Z control of ports by working with TAPA.

25.1.2 Registers

Table 25.2 lists the PIC1-related registers.

For information about restrictions on the individual registers and bits, see the register description in the following sections.

Table 25.2 List of PIC1 registers

<Base: FFDD 0000_H>

Register Name	Symbol	Address
Control register EN	PIC1EN	Base + 00_H
Simultaneous start trigger control register	PIC1SST	Base + 04_H
Simultaneous start control register 2	PIC1SSER2	Base + 18_H
Simultaneous start control register 3	PIC1SSER3	Base + $1C_H$
Hi-Z output control register 2	PIC1HIZCEN2	Base + 88_H
Hi-Z output control register 3	PIC1HIZCEN3	Base + $8C_H$

25.1.2.1 PIC1EN — Control Register EN

This register is set when the PIC1 function is used.

Access: This register can be read/written in 8-bit units.

Address: <Base> + 00_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	PIC1EN0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 25.3 PIC1EN Register Contents

Bit Position	Bit Name	Function
7 to 1	—	Reserved When writing to these bits, write 0.
0	PIC1EN0	Enables and disables the PIC1 function. 0: disabled 1: enabled

25.1.2.2 PIC1SST — Simultaneous Start Trigger Control Register 0

The PIC1SST register generates start triggers of the timers for which simultaneous start is enabled.

Access: This register can be read/written in 8-bit units.

Address: <Base> + 04_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	SYNCTRG
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 25.4 PIC1SST Register Contents

Bit Position	Bit Name	Function
7 to 1	—	Reserved When writing to these bits, write 0.
0	SYNCTRG*1	Generates start triggers of the timers for which simultaneous start is enabled. 0: Invalid 1: Generates a trigger of simultaneous start (outputs a pulse of 1-PCLK-cycle).

Note 1. When this bit is read, the result is always 0.

25.1.2.3 PIC1SSER2 — Simultaneous Start Control Register 2

The PIC1SSER2 register permits start triggers of TSG2n. The selected signal is supplied to the TSTSST signal input line of TSG2n.

Access: This register can be read/written in 16-bit units.

Address: <Base> + 18_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	PIC1SSER209	PIC1SSER208	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R	R	R	R

Table 25.5 PIC1SSER2 Register Contents

Bit Position	Bit Name	Function
15 to 10	—	Reserved When writing to these bits, write 0.
9	PIC1SSER209	Permits TSG21 simultaneous start triggers. 0: TSG21 simultaneous start triggers are not permitted. 1: TSG21 simultaneous start triggers are permitted.
8	PIC1SSER208	Permits TSG20 simultaneous start triggers. 0: TSG20 simultaneous start triggers are not permitted. 1: TSG20 simultaneous start triggers are permitted.
7 to 0	—	Reserved When writing to these bits, write 0.

25.1.2.4 PIC1SSER3 — Simultaneous Start Control Register 3

The PIC1SSER3 register permits start triggers of OSTMn. The selected signal is supplied to the OSTMnTSST signal input line of OSTMn (n = 0 to 2).

Access: This register can be read/written in 16-bit units.

Address: <Base> + $1C_H$

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	PIC1SSER302	PIC1SSER301	PIC1SSER300
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Table 25.6 PIC1SSER3 Register Contents

Bit Position	Bit Name	Function
15 to 3	—	Reserved When writing to these bits, write 0.
2	PIC1SSER302	Permits OSTM2 simultaneous start triggers. 0: OSTM2 simultaneous start triggers are not permitted. 1: OSTM2 simultaneous start triggers are permitted.
1	PIC1SSER301	Permits OSTM1 simultaneous start triggers. 0: OSTM1 simultaneous start triggers are not permitted. 1: OSTM1 simultaneous start triggers are permitted.
0	PIC1SSER300	Permits OSTM0 simultaneous start triggers. 0: OSTM0 simultaneous start triggers are not permitted. 1: OSTM0 simultaneous start triggers are permitted.

25.1.2.5 PIC1HIZCEN2 — Hi-Z Output Control Register 2

The PIC1HIZCEN2 register selects an input signal for Hi-Z output control. The selected signal is supplied to the TAPATHASIN signal input line of TAPA2.

Access: This register can be read/written in 8-bit units.

Address: <Base> + 88_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	PIC1HIZCEN25	—	PIC1HIZCEN23	—	—	PIC1HIZCEN20
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R	R/W	R	R	R/W

Table 25.7 PIC1HIZCEN2 Register Contents

Bit Position	Bit Name	Function
7, 6	—	Reserved When writing to these bits, write 0.
5	PIC1HIZCEN25	Enables or disables Hi-Z output control by the ERROROUTZ signal. 0: Disabled 1: Enabled
4	—	Reserved When writing to these bits, write 0.
3	PIC1HIZCEN23	Enables or disables Hi-Z output control by the INTTSG20IER signal. 0: Disabled 1: Enabled
2 to 1	—	Reserved When writing to these bits, write 0.
0	PIC1HIZCEN20	Enables or disables Hi-Z output control by input to the ESO2 pin. 0: Disabled 1: Enabled

25.1.2.6 PIC1HIZCEN3 — Hi-Z Output Control Register 3

The PIC1HIZCEN3 register selects an input signal for Hi-Z output control. The selected signal is supplied to the TAPATHASIN signal input line of TAPA3.

Access: This register can be read/written in 8-bit units.

Address: <Base> + $8C_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	PIC1HIZCEN35	PIC1HIZCEN34	—	—	—	PIC1HIZCEN30
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R	R	R	R/W

Table 25.8 PIC1HIZCEN3 Register Contents

Bit Position	Bit Name	Function
7 to 6	—	Reserved When writing to these bits, write 0.
5	PIC1HIZCEN35	Enables or disables Hi-Z output control by the ERROROUTZ signal. 0: Disabled 1: Enabled
4	PIC1HIZCEN34	Enables or disables Hi-Z output control by the INTTSG21IER signal. 0: Disabled 1: Enabled
3 to 1	—	Reserved When writing to these bits, write 0.
0	PIC1HIZCEN30	Enables or disables Hi-Z output control by input to the ESO3 pin. 0: Disabled 1: Enabled

25.1.3 Operation

25.1.3.1 Timer Synchronization

(1) An arbitrary combination of timers (TSG2n and OSTMn) that operate the given operating mode is started simultaneously.

(2) Simultaneous start of the target timers is permitted by setting the corresponding bits of PIC1SSER2 and PIC1SSER3 to 1.

(3) Writing 1 to the SYNCTRG bit of PIC1SST0 simultaneously starts the target timers specified in (2).

(4) Repeating (2) and (3) for the channels that are not yet started allow each of the different target timer groups to be started simultaneously in separate timing.

25.2 Trigger Selection Function (PIC2)

25.2.1 Overview

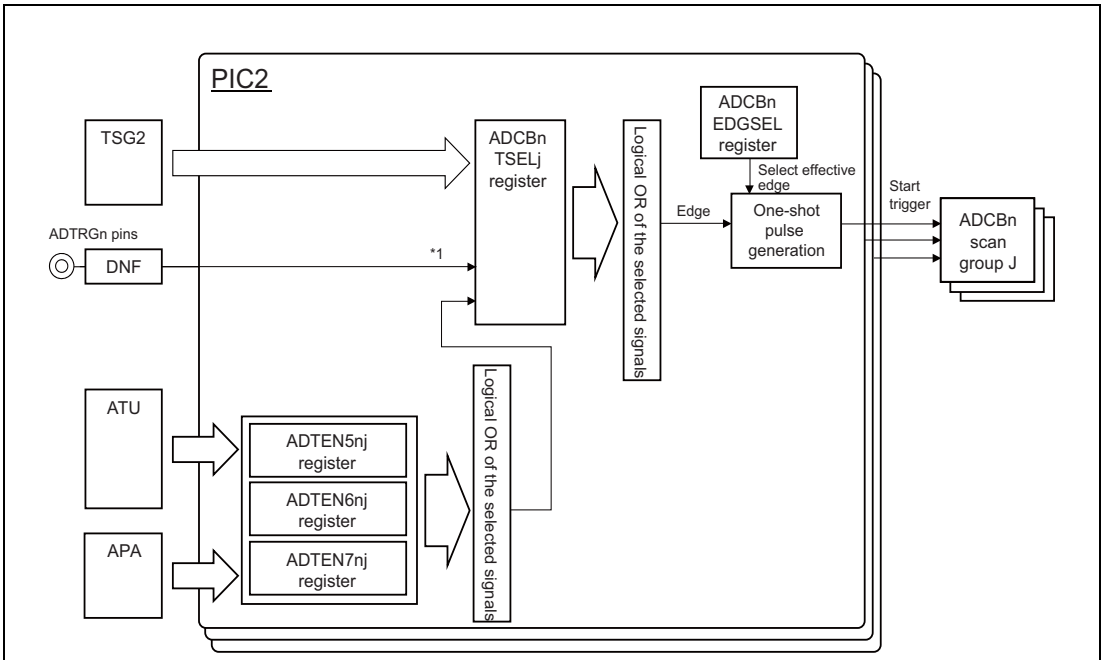
The trigger selection block (PIC2: Peripheral Interconnection-2) selects triggers for the ADC from among signals from the ATU and APA and signals from other internal modules and for starting and stopping ΔΣADC from among internal signals from various timers and other peripheral modules.

Table 25.9 describes the overview of the PIC2 specifications.

Table 25.9 Overview of the PIC2 specifications

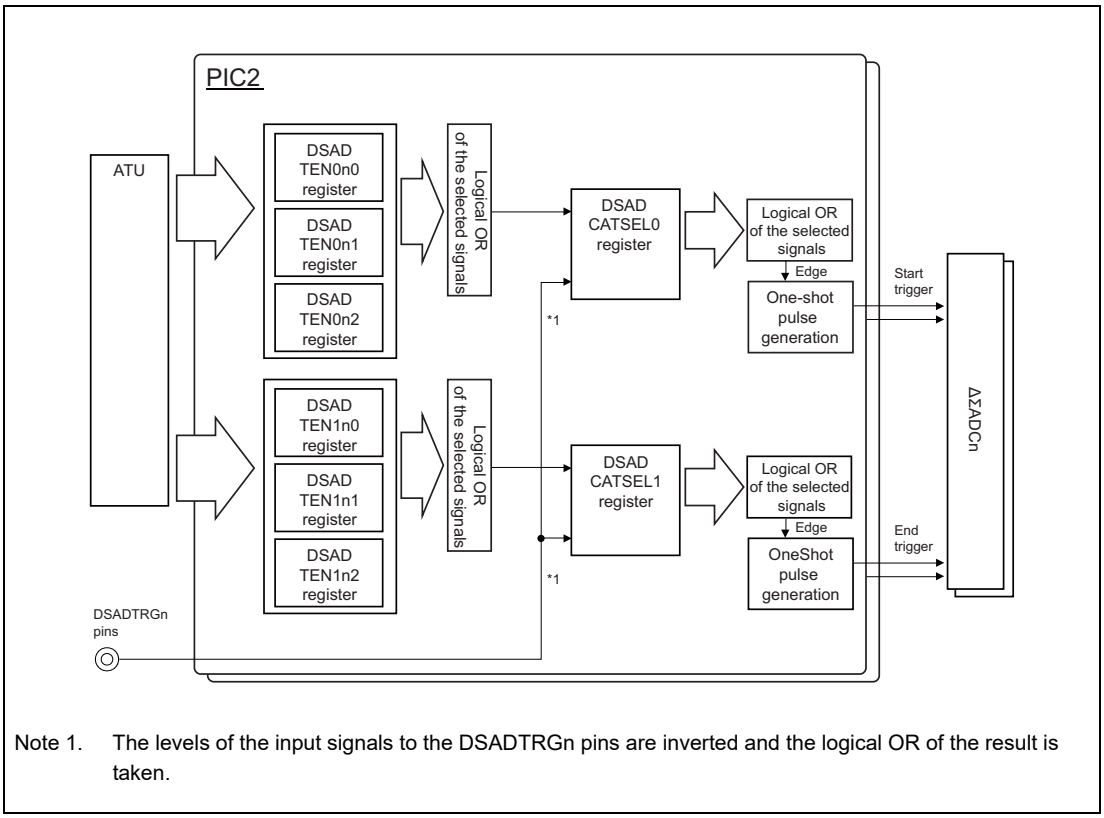
Item	Description
ADC trigger selection function (selection from ATU timers, peripheral IPs, and above described sources)	Selects sources from ATU and APA for ADC.
	Selects an AD trigger from the above selected sources and internal signals from peripheral IPs.
ΔΣADC trigger selection	Selects an AD trigger for ΔΣADC from ATU and external pins.

25.2.2 Block Diagram



Note 1. The levels of the input signals to the ADTRGn pins are inverted and the logical OR of the result is taken.

Figure 25.1 Block Diagram of Trigger Selection Function of ADCBn Scan Group j



Note 1. The levels of the input signals to the DSADTRGn pins are inverted and the logical OR of the result is taken.

Figure 25.2 Block Diagram of Trigger Selection Function of ΔΣADCn

25.2.3 Register

Table 25.10 lists the registers related to PIC2.

For information about restrictions on the individual registers and bits, see the register description in the following sections.

Table 25.10 List of PIC2 Registers (1/3)

<Base: FFDD 1000$_H$>

Register Name	Symbol	Address
ADC trigger selection control register		
A/D converter 0 trigger selection control register 0	PIC2ADCB0TSEL0	Base + 00$_H$
A/D converter 0 trigger selection control register 1	PIC2ADCB0TSEL1	Base + 04$_H$
A/D converter 0 trigger selection control register 2	PIC2ADCB0TSEL2	Base + 08$_H$
A/D converter 0 trigger selection control register 3	PIC2ADCB0TSEL3	Base + 0C$_H$
A/D converter 0 trigger selection control register 4	PIC2ADCB0TSEL4	Base + 10$_H$
A/D converter 0 trigger edge selection control register	PIC2ADCB0EDGSEL	Base + 1C$_H$
A/D converter 1 trigger selection control register 0	PIC2ADCB1TSEL0	Base + 20$_H$
A/D converter 1 trigger selection control register 1	PIC2ADCB1TSEL1	Base + 24$_H$
A/D converter 1 trigger selection control register 2	PIC2ADCB1TSEL2	Base + 28$_H$
A/D converter 1 trigger selection control register 3	PIC2ADCB1TSEL3	Base + 2C$_H$
A/D converter 1 trigger selection control register 4	PIC2ADCB1TSEL4	Base + 30$_H$
A/D converter 1 trigger edge selection control register	PIC2ADCB1EDGSEL	Base + 3C$_H$
AD converter trigger output control register 500	PIC2ADTEN500	Base + 80$_H$
AD converter trigger output control register 501	PIC2ADTEN501	Base + 84$_H$
AD converter trigger output control register 502	PIC2ADTEN502	Base + 88$_H$
AD converter trigger output control register 503	PIC2ADTEN503	Base + 8C$_H$
AD converter trigger output control register 504	PIC2ADTEN504	Base + 90$_H$
AD converter trigger output control register 510	PIC2ADTEN510	Base + 94$_H$
AD converter trigger output control register 511	PIC2ADTEN511	Base + 98$_H$
AD converter trigger output control register 512	PIC2ADTEN512	Base + 9C$_H$
AD converter trigger output control register 513	PIC2ADTEN513	Base + A0$_H$
AD converter trigger output control register 514	PIC2ADTEN514	Base + A4$_H$
AD converter trigger output control register 600	PIC2ADTEN600	Base + A8$_H$
AD converter trigger output control register 601	PIC2ADTEN601	Base + AC$_H$
AD converter trigger output control register 602	PIC2ADTEN602	Base + B0$_H$
AD converter trigger output control register 603	PIC2ADTEN603	Base + B4$_H$
AD converter trigger output control register 604	PIC2ADTEN604	Base + B8$_H$
AD converter trigger output control register 610	PIC2ADTEN610	Base + BC$_H$
AD converter trigger output control register 611	PIC2ADTEN611	Base + C0$_H$
AD converter trigger output control register 612	PIC2ADTEN612	Base + C4$_H$
AD converter trigger output control register 613	PIC2ADTEN613	Base + C8$_H$
AD converter trigger output control register 614	PIC2ADTEN614	Base + CC$_H$
A/D converter trigger output control register 700	PIC2ADTEN700	Base + D0$_H$
A/D converter trigger output control register 701	PIC2ADTEN701	Base + D4$_H$
A/D converter trigger output control register 702	PIC2ADTEN702	Base + D8$_H$
A/D converter trigger output control register 703	PIC2ADTEN703	Base + DC$_H$
A/D converter trigger output control register 704	PIC2ADTEN704	Base + E0$_H$

Table 25.10 List of PIC2 Registers (2/3)

<Base: FFDD 1000$_H$>

Register Name	Symbol	Address
A/D converter trigger output control register 710	PIC2ADTEN710	Base + E4$_H$
A/D converter trigger output control register 711	PIC2ADTEN711	Base + E8$_H$
A/D converter trigger output control register 712	PIC2ADTEN712	Base + EC$_H$
A/D converter trigger output control register 713	PIC2ADTEN713	Base + F0$_H$
A/D converter trigger output control register 714	PIC2ADTEN714	Base + F4$_H$
DS-ADC trigger selection control register		
DSADC start trigger output control register 00	PIC2DSADTEN000	Base + 100$_H$
DSADC start trigger output control register 01	PIC2DSADTEN001	Base + 104$_H$
DSADC start trigger output control register 02	PIC2DSADTEN002	Base + 108$_H$
DSADC stop trigger output control register 00	PIC2DSADTEN100	Base + 10C$_H$
DSADC stop trigger output control register 01	PIC2DSADTEN101	Base + 11C$_H$
DSADC stop trigger output control register 02	PIC2DSADTEN102	Base + 114$_H$
DSADC start trigger output control register 10	PIC2DSADTEN010	Base + 118$_H$
DSADC start trigger output control register 11	PIC2DSADTEN011	Base + 11C$_H$
DSADC start trigger output control register 12	PIC2DSADTEN012	Base + 120$_H$
DSADC stop trigger output control register 10	PIC2DSADTEN110	Base + 124$_H$
DSADC stop trigger output control register 11	PIC2DSADTEN111	Base + 128$_H$
DSADC stop trigger output control register 12	PIC2DSADTEN112	Base + 12C$_H$
DSADC start trigger output control register 20	PIC2DSADTEN020	Base + 130$_H$
DSADC start trigger output control register 21	PIC2DSADTEN021	Base + 134$_H$
DSADC start trigger output control register 22	PIC2DSADTEN022	Base + 138$_H$
DSADC stop trigger output control register 20	PIC2DSADTEN120	Base + 13C$_H$
DSADC stop trigger output control register 21	PIC2DSADTEN121	Base + 140$_H$
DSADC stop trigger output control register 22	PIC2DSADTEN122	Base + 144$_H$
DSADC start trigger output control register 30	PIC2DSADTEN030	Base + 148$_H$
DSADC start trigger output control register 31	PIC2DSADTEN031	Base + 14C$_H$
DSADC start trigger output control register 32	PIC2DSADTEN032	Base + 150$_H$
DSADC stop trigger output control register 30	PIC2DSADTEN130	Base + 154$_H$
DSADC stop trigger output control register 31	PIC2DSADTEN131	Base + 158$_H$
DSADC stop trigger output control register 32	PIC2DSADTEN132	Base + 15C$_H$
DSADC start trigger output control register 40	PIC2DSADTEN040	Base + 160$_H$
DSADC start trigger output control register 41	PIC2DSADTEN041	Base + 164$_H$
DSADC start trigger output control register 42	PIC2DSADTEN042	Base + 168$_H$
DSADC stop trigger output control register 40	PIC2DSADTEN140	Base + 16C$_H$
DSADC stop trigger output control register 41	PIC2DSADTEN141	Base + 170$_H$
DSADC stop trigger output control register 42	PIC2DSADTEN142	Base + 174$_H$
DSADC start trigger output control register 50	PIC2DSADTEN050	Base + 178$_H$
DSADC start trigger output control register 51	PIC2DSADTEN051	Base + 17C$_H$
DSADC start trigger output control register 52	PIC2DSADTEN052	Base + 180$_H$
DSADC stop trigger output control register 50	PIC2DSADTEN150	Base + 184$_H$
DSADC stop trigger output control register 51	PIC2DSADTEN151	Base + 188$_H$
DSADC stop trigger output control register 52	PIC2DSADTEN152	Base + 18C$_H$
DSADC start trigger output control register 60	PIC2DSADTEN060	Base + 190$_H$

Table 25.10 List of PIC2 Registers (3/3)

<Base: FFDD 1000_H>

Register Name	Symbol	Address
DSADC start trigger output control register 61	PIC2DSADTEN061	Base + 194_H
DSADC start trigger output control register 62	PIC2DSADTEN062	Base + 198_H
DSADC stop trigger output control register 60	PIC2DSADTEN160	Base + $19C_H$
DSADC stop trigger output control register 61	PIC2DSADTEN161	Base + $1A0_H$
DSADC stop trigger output control register 62	PIC2DSADTEN162	Base + $1A4_H$
DSADC start trigger output control register 70	PIC2DSADTEN070	Base + $1A8_H$
DSADC start trigger output control register 71	PIC2DSADTEN071	Base + $1AC_H$
DSADC start trigger output control register 72	PIC2DSADTEN072	Base + $1B0_H$
DSADC stop trigger output control register 70	PIC2DSADTEN170	Base + $1B4_H$
DSADC stop trigger output control register 71	PIC2DSADTEN171	Base + $1B8_H$
DSADC stop trigger output control register 72	PIC2DSADTEN172	Base + $1BC_H$
DSADC converter trigger selection control register 0	PIC2DSADCATSEL0	Base + $1C0_H$
DSADC converter trigger selection control register 1	PIC2DSADCATSEL1	Base + $1C4_H$

25.2.3.1 PIC2ADCBnTSELj — AD Converter n Trigger Selection Control Register j

The PIC2ADCBnTSELj register selects a trigger for ADCBn scan group j (n = 0, 1; j = 0 to 4).

Access: This register can be read/written in 16- or 32-bit units.

Address: <Base> + 00_H (n = 0, j = 0), <Base> + 04_H (n = 0, j = 1),
<Base> + 08_H (n = 0, j = 2), <Base> + $0C_H$ (n = 0, j = 3),
<Base> + 10_H (n = 0, j = 4), <Base> + 20_H (n = 1, j = 0),
<Base> + 24_H (n = 1, j = 1), <Base> + 28_H (n = 1, j = 2),
<Base> + $2C_H$ (n = 1, j = 3), <Base> + 30_H (n = 1, j = 4)

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PIC2ADCBnTSELj15	—	—	—	—	—	—	PIC2ADCBnTSELj08	PIC2ADCBnTSELj07	PIC2ADCBnTSELj06	PIC2ADCBnTSELj05	PIC2ADCBnTSELj04	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R

Table 25.11 PIC2ADCBnTSELj Register Contents

Bit Position	Bit Name	Function
31 to 16	—	Reserved When writing to these bits, write 0.
15	PIC2ADCBnTSELj15	Selects the trigger selected by the PIC2ADTEN5nj, PIC2ADTEN6nj, or PIC2ADTEN7nj register as a trigger source of ADCBn scan group j. 0: Does not select the trigger selected by register PIC2ADTEN5nj, 6nj, or 7nj. 1: Selects the trigger selected by register PIC2ADTEN5nj, 6nj, or 7nj.
14 to 9	—	Reserved When writing to these bits, write 0.
8	PIC2ADCBnTSELj08	Selects the ADTRGn pin as a trigger source of ADCBn scan group j. 0: The ADTRGn pin is not selected. 1: The ADTRGn pin is selected.
7	PIC2ADCBnTSELj07	Selects TSG21TSTADT1 as a trigger source of ADCBn scan group j. (TSG21 TSTADT1 signal) 0: TSG21TSTADT1 is not selected. 1: TSG21TSTADT1 is selected.
6	PIC2ADCBnTSELj06	Selects TSG21TSTADT0 as a trigger source of ADCBn scan group j. (TSG21 TSTADT0 signal) 0: TSG21TSTADT0 is not selected. 1: TSG21TSTADT0 is selected.
5	PIC2ADCBnTSELj05	Selects TSG20TSTADT1 as a trigger source of ADCBn scan group j. (TSG20 TSTADT1 signal) 0: TSG20TSTADT1 is not selected. 1: TSG20TSTADT1 is selected.
4	PIC2ADCBnTSELj04	Selects TSG20TSTADT0 as a trigger source of ADCBn scan group j. (TSG20 TSTADT0 signal) 0: TSG20TSTADT0 is not selected. 1: TSG20TSTADT0 is selected.
3 to 0	—	Reserved When writing to these bits, write 0.

25.2.3.2 PIC2ADCBnEDGSEL — AD Converter n Trigger Edge Selection Control Register

The PIC2ADCBnEDGSEL register selects an effective edge for the one-shot pulse generation circuit which generates an ADCB trigger.

The ADC external pin trigger is input in negative logic, but it is converted to positive logic for trigger source selection. Since edge detection is made for a trigger source after selection, note that the definition of an edge is reversed for an ADC external pin signal. (Setting 00 enables selection of a falling edge of ADC external pin triggers ADTRG0 and ADTRG1).

Access: This register can be read in 16-bit units.

Address: <Base> + $1C_H$ (n = 0), <Base> + $3C_H$ (n = 1)

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	PIC2ADCBnEDGSEL98[9:8]		PIC2ADCBnEDGSEL76[7:6]		PIC2ADCBnEDGSEL54[5:4]		PIC2ADCBnEDGSEL32[3:2]		PIC2ADCBnEDGSEL10[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.12 PIC2ADCBnEDGSEL Register Contents

Bit Position	Bit Name	Function
15 to 10	—	Reserved When writing to these bits, write 0.
9, 8	PIC2ADCBnEDGSEL98[9:8]	Select an effective edge of ADCBn scan group 4. 00: Rising edge is selected. 01: Falling edge is selected. 10: Both edges are selected. 11: — (setting prohibited)
7, 6	PIC2ADCBnEDGSEL76[7:6]	Select an effective edge of ADCBn scan group 3. 00: Rising edge is selected. 01: Falling edge is selected. 10: Both edges are selected. 11: — (setting prohibited)
5, 4	PIC2ADCBnEDGSEL54[5:4]	Select an effective edge of ADCBn scan group 2. 00: Rising edge is selected. 01: Falling edge is selected. 10: Both edges are selected. 11: — (setting prohibited)
3, 2	PIC2ADCBnEDGSEL32[3:2]	Select an effective edge of ADCBn scan group 1. 00: Rising edge is selected. 01: Falling edge is selected. 10: Both edges are selected. 11: — (setting prohibited)
1, 0	PIC2ADCBnEDGSEL10[1:0]	Select an effective edge of ADCBn scan group 0. 00: Rising edge is selected. 01: Falling edge is selected. 10: Both edges are selected. 11: — (setting prohibited)

25.2.3.3 PIC2ADTEN5nj — AD Converter Trigger Output Control Register 5nj

The PIC2ADTEN5nj register enables a trigger source from ATU timer C compare match, ATU timer G compare match, and ATU DMA request switch function to be selected as an ADCB trigger (n = 0, 1; j = 0 to 4).

Access: This register can be read/written in 16- or 32-bit units.

Address: <Base> + 80_H (n = 0, j = 0), <Base> + 84_H (n = 0, j = 1),
<Base> + 88_H (n = 0, j = 2), <Base> + $8C_H$ (n = 0, j = 3),
<Base> + 90_H (n = 0, j = 4), <Base> + 94_H (n = 1, j = 0),
<Base> + 98_H (n = 1, j = 1), <Base> + $9C_H$ (n = 1, j = 2),
<Base> + $A0_H$ (n = 1, j = 3), <Base> + $A4_H$ (n = 1, j = 4)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PIC2ADTEN5nj31	PIC2ADTEN5nj30	PIC2ADTEN5nj29	PIC2ADTEN5nj28	PIC2ADTEN5nj27	PIC2ADTEN5nj26	PIC2ADTEN5nj25	PIC2ADTEN5nj24	PIC2ADTEN5nj23	PIC2ADTEN5nj22	PIC2ADTEN5nj21	PIC2ADTEN5nj20	PIC2ADTEN5nj19	PIC2ADTEN5nj18	PIC2ADTEN5nj17	PIC2ADTEN5nj16
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PIC2ADTEN5nj15	PIC2ADTEN5nj14	PIC2ADTEN5nj13	PIC2ADTEN5nj12	PIC2ADTEN5nj11	PIC2ADTEN5nj10	PIC2ADTEN5nj9	PIC2ADTEN5nj8	—	—	—	—	—	—	—	PIC2ADTEN5nj0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R/W

Table 25.13 PIC2ADTEN5nj Register Contents (1/3)

Bit Position	Bit Name	Function
31	PIC2ADTEN5nj31	OCRC73 input capture/compare match interrupt (ATU4 timer C subblock 7 channel 3 input capture/compare match interrupt) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.
30	PIC2ADTEN5nj30	OCRC72 input capture/compare match interrupt (ATU4 timer C subblock 7 channel 2 input capture/compare match interrupt) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.
29	PIC2ADTEN5nj29	OCRC71 input capture/compare match interrupt (ATU4 timer C subblock 7 channel 1 input capture/compare match interrupt) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.
28	PIC2ADTEN5nj28	OCRC70 input capture/compare match interrupt (ATU4 timer C subblock 7 channel 0 input capture/compare match interrupt) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.
27	PIC2ADTEN5nj27	OCRC63 input capture/compare match interrupt (ATU4 timer C subblock 6 channel 3 input capture/compare match interrupt) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.
26	PIC2ADTEN5nj26	OCRC62 input capture/compare match interrupt (ATU4 timer C subblock 6 channel 2 input capture/compare match interrupt) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.
25	PIC2ADTEN5nj25	OCRC61 input capture/compare match interrupt (ATU4 timer C subblock 6 channel 1 input capture/compare match interrupt) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.

Table 25.13 PIC2ADTEN5nj Register Contents (2/3)

Bit Position	Bit Name	Function
24	PIC2ADTEN5nj24	OCRC60 input capture/compare match interrupt (ATU4 timer C subblock 6 channel 0 input capture/compare match interrupt) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.
23	PIC2ADTEN5nj23	OCRC43 input capture/compare match interrupt (ATU4 timer C subblock 4 channel 3 input capture/compare match interrupt) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.
22	PIC2ADTEN5nj22	OCRC42 input capture/compare match interrupt (ATU4 timer C subblock 4 channel 2 input capture/compare match interrupt) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.
21	PIC2ADTEN5nj21	OCRC41 input capture/compare match interrupt (ATU4 timer C subblock 4 channel 1 input capture/compare match interrupt) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.
20	PIC2ADTEN5nj20	OCRC40 input capture/compare match interrupt (ATU4 timer C subblock 4 channel 0 input capture/compare match interrupt) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.
19	PIC2ADTEN5nj19	OCRC03 input capture/compare match interrupt (ATU4 timer C subblock 0 channel 3 input capture/compare match interrupt) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.
18	PIC2ADTEN5nj18	OCRC02 input capture/compare match interrupt (ATU4 timer C subblock 0 channel 2 input capture/compare match interrupt) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.
17	PIC2ADTEN5nj17	OCRC01 input capture/compare match interrupt (ATU4 timer C subblock 0 channel 1 input capture/compare match interrupt) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.
16	PIC2ADTEN5nj16	OCRC00 input capture/compare match interrupt (ATU4 timer C subblock 0 channel 0 input capture/compare match interrupt) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.
15	PIC2ADTEN5nj15	OCRG7 compare match interrupt (ATU4 timer G subblock 7) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.
14	PIC2ADTEN5nj14	OCRG6 compare match interrupt (ATU4 timer G subblock 6) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.
13	PIC2ADTEN5nj13	OCRG5 compare match interrupt (ATU4 timer G subblock 5) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.
12	PIC2ADTEN5nj12	OCRG4 compare match interrupt (ATU4 timer G subblock 4) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.
11	PIC2ADTEN5nj11	OCRG3 compare match interrupt (ATU4 timer G subblock 3) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.
10	PIC2ADTEN5nj10	OCRG2 compare match interrupt (ATU4 timer G subblock 2) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.
9	PIC2ADTEN5nj9	OCRG1 compare match interrupt (ATU4 timer G subblock 1) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.

Table 25.13 PIC2ADTEN5nj Register Contents (3/3)

Bit Position	Bit Name	Function
8	PIC2ADTEN5nj8	OCRG0 compare match interrupt (ATU4 timer G subblock 0) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.
7 to 1	—	Reserved When writing to these bits, write 0.
0	PIC2ADTEN5nj0	Output signal from the SAR-AD trigger selection circuit for ATU DMA/AD request auto-switching 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.

25.2.3.4 PIC2ADTEN6nj — AD Converter Trigger Output Control Register 6nj

The PIC2ADTEN6nj register enables a trigger source from APA0 interrupt and ATU timer D compare match A to be selected as an ADCB trigger (n = 0, 1; j = 0 to 4).

Access: This register can be read/written in 32-bit units.

Address: <Base> + $A8_H$ (n = 0, j = 0), <Base> + AC_H (n = 0, j = 1),
<Base> + $B0_H$ (n = 0, j = 2), <Base> + $B4_H$ (n = 0, j = 3),
<Base> + $B8_H$ (n = 0, j = 4), <Base> + BC_H (n = 1, j = 0),
<Base> + $C0_H$ (n = 1, j = 1), <Base> + $C4_H$ (n = 1, j = 2),
<Base> + $C8_H$ (n = 1, j = 3), <Base> + CC_H (n = 1, j = 4)

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PIC2ADTEN6nj31	PIC2ADTEN6nj30	PIC2ADTEN6nj29	PIC2ADTEN6nj28	—	—	—	—	—	—	—	—	PIC2ADTEN6nj19	PIC2ADTEN6nj18	PIC2ADTEN6nj17	PIC2ADTEN6nj16
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PIC2ADTEN6nj15	PIC2ADTEN6nj14	PIC2ADTEN6nj13	PIC2ADTEN6nj12	PIC2ADTEN6nj11	PIC2ADTEN6nj10	PIC2ADTEN6nj9	PIC2ADTEN6nj8	PIC2ADTEN6nj7	PIC2ADTEN6nj6	PIC2ADTEN6nj5	PIC2ADTEN6nj4	PIC2ADTEN6nj3	PIC2ADTEN6nj2	PIC2ADTEN6nj1	PIC2ADTEN6nj0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.14 PIC2ADTEN6nj Register Contents (1/3)

Bit Position	Bit Name	Function
31	PIC2ADTEN6nj31	APA0 PWM output ch.3 status change interrupt 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.
30	PIC2ADTEN6nj30	APA0 PWM output ch.2 status change interrupt 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.
29	PIC2ADTEN6nj29	APA0 PWM output ch.1 status change interrupt 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.
28	PIC2ADTEN6nj28	APA0 PWM output ch.0 status change interrupt 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.
27 to 20	—	Reserved When writing to these bits, always write 0.
19	PIC2ADTEN6nj19	OCR1D93 compare match interrupt (ATU4 timer D subblock 9 channel 3 compare match A interrupt) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.
18	PIC2ADTEN6nj18	OCR1D92 compare match interrupt (ATU4 timer D subblock 9 channel 2 compare match A interrupt) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.
17	PIC2ADTEN6nj17	OCR1D91 compare match interrupt (ATU4 timer D subblock 9 channel 1 compare match A interrupt) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.
16	PIC2ADTEN6nj16	OCR1D90 compare match interrupt (ATU4 timer D subblock 9 channel 0 compare match A interrupt) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.

Table 25.14 PIC2ADTEN6nj Register Contents (2/3)

Bit Position	Bit Name	Function
15	PIC2ADTEN6nj15	OCR1D63 compare match interrupt (ATU4 timer D subblock 6 channel 3 compare match A interrupt) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.
14	PIC2ADTEN6nj14	OCR1D62 compare match interrupt (ATU4 timer D subblock 6 channel 2 compare match A interrupt) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.
13	PIC2ADTEN6nj13	OCR1D61 compare match interrupt (ATU4 timer D subblock 6 channel 1 compare match A interrupt) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.
12	PIC2ADTEN6nj12	OCR1D60 compare match interrupt (ATU4 timer D subblock 6 channel 0 compare match A interrupt) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.
11	PIC2ADTEN6nj11	OCR1D43 compare match interrupt (ATU4 timer D subblock 4 channel 3 compare match A interrupt) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.
10	PIC2ADTEN6nj10	OCR1D42 compare match interrupt (ATU4 timer D subblock 4 channel 2 compare match A interrupt) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.
9	PIC2ADTEN6nj9	OCR1D41 compare match interrupt (ATU4 timer D subblock 4 channel 1 compare match A interrupt) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.
8	PIC2ADTEN6nj8	OCR1D40 compare match interrupt (ATU4 timer D subblock 4 channel 0 compare match A interrupt) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.
7	PIC2ADTEN6nj7	OCR1D23 compare match interrupt (ATU4 timer D subblock 2 channel 3 compare match A interrupt) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.
6	PIC2ADTEN6nj6	OCR1D22 compare match interrupt (ATU4 timer D subblock 2 channel 2 compare match A interrupt) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.
5	PIC2ADTEN6nj5	OCR1D21 compare match interrupt (ATU4 timer D subblock 2 channel 1 compare match A interrupt) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.
4	PIC2ADTEN6nj4	OCR1D20 compare match interrupt (ATU4 timer D subblock 2 channel 0 compare match A interrupt) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.
3	PIC2ADTEN6nj3	OCR1D03 compare match interrupt (ATU4 timer D subblock 0 channel 3 compare match A interrupt) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.
2	PIC2ADTEN6nj2	OCR1D02 compare match interrupt (ATU4 timer D subblock 0 channel 2 compare match A interrupt) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.

Table 25.14 PIC2ADTEN6nj Register Contents (3/3)

Bit Position	Bit Name	Function
1	PIC2ADTEN6nj1	OCR1D01 compare match interrupt (ATU4 timer D subblock 0 channel 1 compare match A interrupt) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.
0	PIC2ADTEN6nj0	OCR1D00 compare match interrupt (ATU4 timer D subblock 0 channel 0 compare match A interrupt) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.

25.2.3.5 PIC2ADTEN7nj — A/D Converter Trigger Output Control Register 7nj

The PIC2ADTEN7nj register enables a trigger source from ATU timer D compare match B to be selected as an ADCB trigger (n = 0, 1; j = 0 to 4).

Access: This register can be read/written in 32-bit units.

Address: <Base> + $D0_H$ (n = 0, j = 0), <Base> + $D4_H$ (n = 0, j = 1), <Base> + $D8_H$ (n = 0, j = 2), <Base> + DC_H (n = 0, j = 3), <Base> + $E0_H$ (n = 0, j = 4), <Base> + $E4_H$ (n = 1, j = 0), <Base> + $E8_H$ (n = 1, j = 1), <Base> + EC_H (n = 1, j = 2), <Base> + $F0_H$ (n = 1, j = 3), <Base> + $F4_H$ (n = 1, j = 4)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	PIC2ADTEN7nj19	PIC2ADTEN7nj18	PIC2ADTEN7nj17	PIC2ADTEN7nj16
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PIC2ADTEN7nj15	PIC2ADTEN7nj14	PIC2ADTEN7nj13	PIC2ADTEN7nj12	PIC2ADTEN7nj11	PIC2ADTEN7nj10	PIC2ADTEN7nj9	PIC2ADTEN7nj8	PIC2ADTEN7nj7	PIC2ADTEN7nj6	PIC2ADTEN7nj5	PIC2ADTEN7nj4	PIC2ADTEN7nj3	PIC2ADTEN7nj2	PIC2ADTEN7nj1	PIC2ADTEN7nj0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.15 PIC2ADTEN7nj Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 20	—	Reserved When writing to these bits, write 0.
19	PIC2ADTEN7nj19	OCR2D93 compare match interrupt (ATU4 timer D subblock 9 channel 3 compare match B interrupt) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.
18	PIC2ADTEN7nj18	OCR2D92 compare match interrupt (ATU4 timer D subblock 9 channel 2 compare match B interrupt) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.
17	PIC2ADTEN7nj17	OCR2D91 compare match interrupt (ATU4 timer D subblock 9 channel 1 compare match B interrupt) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.
16	PIC2ADTEN7nj16	OCR2D90 compare match interrupt (ATU4 timer D subblock 9 channel 0 compare match B interrupt) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.
15	PIC2ADTEN7nj15	OCR2D63 compare match interrupt (ATU4 timer D subblock 6 channel 3 compare match B interrupt) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.
14	PIC2ADTEN7nj14	OCR2D62 compare match interrupt (ATU4 timer D subblock 6 channel 2 compare match B interrupt) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.
13	PIC2ADTEN7nj13	OCR2D61 compare match interrupt (ATU4 timer D subblock 6 channel 1 compare match B interrupt) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.

Table 25.15 PIC2ADTEN7nj Register Contents (2/2)

Bit Position	Bit Name	Function
12	PIC2ADTEN7nj12	OCR2D60 compare match interrupt (ATU4 timer D subblock 6 channel 0 compare match B interrupt) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.
11	PIC2ADTEN7nj11	OCR2D43 compare match interrupt (ATU4 timer D subblock 4 channel 3 compare match B interrupt) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.
10	PIC2ADTEN7nj10	OCR2D42 compare match interrupt (ATU4 timer D subblock 4 channel 2 compare match B interrupt) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.
9	PIC2ADTEN7nj9	OCR2D41 compare match interrupt (ATU4 timer D subblock 4 channel 1 compare match B interrupt) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.
8	PIC2ADTEN7nj8	OCR2D40 compare match interrupt (ATU4 timer D subblock 4 channel 0 compare match B interrupt) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.
7	PIC2ADTEN7nj7	OCR2D23 compare match interrupt (ATU4 timer D subblock 2 channel 3 compare match B interrupt) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.
6	PIC2ADTEN7nj6	OCR2D22 compare match interrupt (ATU4 timer D subblock 2 channel 2 compare match B interrupt) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.
5	PIC2ADTEN7nj5	OCR2D21 compare match interrupt (ATU4 timer D subblock 2 channel 1 compare match B interrupt) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.
4	PIC2ADTEN7nj4	OCR2D20 compare match interrupt (ATU4 timer D subblock 2 channel 0 compare match B interrupt) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.
3	PIC2ADTEN7nj3	OCR2D03 compare match interrupt (ATU4 timer D subblock 0 channel 3 compare match B interrupt) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.
2	PIC2ADTEN7nj2	OCR2D02 compare match interrupt (ATU4 timer D subblock 0 channel 2 compare match B interrupt) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.
1	PIC2ADTEN7nj1	OCR2D01 compare match interrupt (ATU4 timer D subblock 0 channel 1 compare match B interrupt) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.
0	PIC2ADTEN7nj0	OCR2D00 compare match interrupt (ATU4 timer D subblock 0 channel 0 compare match B interrupt) 0: Selection of the above signal as an ADCB trigger is disabled. 1: Selection of the above signal as an ADCB trigger is enabled.

25.2.3.6 PIC2DSADTEN0n0 — DSADC Start Trigger Output Control Register n0 (n = 0 to 7)

DSADC start trigger output control register n0 enables a trigger source from ATU timer G compare match to be selected as a trigger for starting ΔΣADC.

Access: This register can be read/written in 16-bit units.

Address: <Base> + 100_H (n = 0), <Base> + 118_H (n = 1),
<Base> + 130_H (n = 2), <Base> + 148_H (n = 3),
<Base> + 160_H (n = 4), <Base> + 178_H (n = 5),
<Base> + 190_H (n = 6), <Base> + $1A8_H$ (n = 7)

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PIC2DSADTEN 0n015	PIC2DSADTEN 0n014	PIC2DSADTEN 0n013	PIC2DSADTEN 0n012	PIC2DSADTEN 0n011	PIC2DSADTEN 0n010	PIC2DSADTEN 0n09	PIC2DSADTEN 0n08	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Table 25.16 PIC2DSADTEN0n0 Register Contents

Bit Position	Bit Name	Function
15	PIC2DSADTEN 0n015	OCRG7 compare match interrupt (ATU4 timer G subblock 7) 0: Selection of the above signal as a ΔΣADCn start trigger is disabled. 1: Selection of the above signal as a ΔΣADCn start trigger is enabled.
14	PIC2DSADTEN 0n014	OCRG6 compare match interrupt (ATU4 timer G subblock 6) 0: Selection of the above signal as a ΔΣADCn start trigger is disabled. 1: Selection of the above signal as a ΔΣADCn start trigger is enabled.
13	PIC2DSADTEN 0n013	OCRG5 compare match interrupt (ATU4 timer G subblock 5) 0: Selection of the above signal as a ΔΣADCn start trigger is disabled. 1: Selection of the above signal as a ΔΣADCn start trigger is enabled.
12	PIC2DSADTEN 0n012	OCRG4 compare match interrupt (ATU4 timer G subblock 4) 0: Selection of the above signal as a ΔΣADCn start trigger is disabled. 1: Selection of the above signal as a ΔΣADCn start trigger is enabled.
11	PIC2DSADTEN 0n011	OCRG3 compare match interrupt (ATU4 timer G subblock 3) 0: Selection of the above signal as a ΔΣADCn start trigger is disabled. 1: Selection of the above signal as a ΔΣADCn start trigger is enabled.
10	PIC2DSADTEN 0n010	OCRG2 compare match interrupt (ATU4 timer G subblock 2) 0: Selection of the above signal as a ΔΣADCn start trigger is disabled. 1: Selection of the above signal as a ΔΣADCn start trigger is enabled.
9	PIC2DSADTEN 0n09	OCRG1 compare match interrupt (ATU4 timer G subblock 1) 0: Selection of the above signal as a ΔΣADCn start trigger is disabled. 1: Selection of the above signal as a ΔΣADCn start trigger is enabled.
8	PIC2DSADTEN 0n08	OCRG0 compare match interrupt (ATU4 timer G subblock 0) 0: Selection of the above signal as a ΔΣADCn start trigger is disabled. 1: Selection of the above signal as a ΔΣADCn start trigger is enabled.
7 to 0	—	Reserved When writing to these bits, write 0.

25.2.3.7 PIC2DSADTEN0n1 — DSADC Start Trigger Output Control Register n1 (n = 0 to 7)

DSADC start trigger output control register n1 enables a trigger source from ATU timer D compare match A to be selected as a trigger for starting ΔΣADC.

Access: This register can be read/written in 32-bit units.

Address: <Base> + 104_H (n = 0), <Base> + $11C_H$ (n = 1), <Base> + 134_H (n = 2), <Base> + $14C_H$ (n = 3), <Base> + 164_H (n = 4), <Base> + $17C_H$ (n = 5), <Base> + 194_H (n = 6), <Base> + $1AC_H$ (n = 7)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	PIC2DSADTEN 0n119	PIC2DSADTEN 0n118	PIC2DSADTEN 0n117	PIC2DSADTEN 0n116
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PIC2DSADTEN 0n115	PIC2DSADTEN 0n114	PIC2DSADTEN 0n113	PIC2DSADTEN 0n112	PIC2DSADTEN 0n111	PIC2DSADTEN 0n110	PIC2DSADTEN 0n19	PIC2DSADTEN 0n18	PIC2DSADTEN 0n17	PIC2DSADTEN 0n16	PIC2DSADTEN 0n15	PIC2DSADTEN 0n14	PIC2DSADTEN 0n13	PIC2DSADTEN 0n12	PIC2DSADTEN 0n11	PIC2DSADTEN 0n10
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.17 PIC2DSADTEN0n1 Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 20	—	Reserved When writing to these bits, write 0.
19	PIC2DSADTEN 0n119	OCR1D93 compare match interrupt (ATU4 timer D subblock 9 channel 3 compare match A interrupt) 0: Selection of the above signal as a ΔΣADCn start trigger is disabled. 1: Selection of the above signal as a ΔΣADCn start trigger is enabled.
18	PIC2DSADTEN 0n118	OCR1D92 compare match interrupt (ATU4 timer D subblock 9 channel 2 compare match A interrupt) 0: Selection of the above signal as a ΔΣADCn start trigger is disabled. 1: Selection of the above signal as a ΔΣADCn start trigger is enabled.
17	PIC2DSADTEN 0n117	OCR1D91 compare match interrupt (ATU4 timer D subblock 9 channel 1 compare match A interrupt) 0: Selection of the above signal as a ΔΣADCn start trigger is disabled. 1: Selection of the above signal as a ΔΣADCn start trigger is enabled.
16	PIC2DSADTEN 0n116	OCR1D90 compare match interrupt (ATU4 timer D subblock 9 channel 0 compare match A interrupt) 0: Selection of the above signal as a ΔΣADCn start trigger is disabled. 1: Selection of the above signal as a ΔΣADCn start trigger is enabled.
15	PIC2DSADTEN 0n115	OCR1D63 compare match interrupt (ATU4 timer D subblock 6 channel 3 compare match A interrupt) 0: Selection of the above signal as a ΔΣADCn start trigger is disabled. 1: Selection of the above signal as a ΔΣADCn start trigger is enabled.
14	PIC2DSADTEN 0n114	OCR1D62 compare match interrupt (ATU4 timer D subblock 6 channel 2 compare match A interrupt) 0: Selection of the above signal as a ΔΣADCn start trigger is disabled. 1: Selection of the above signal as a ΔΣADCn start trigger is enabled.
13	PIC2DSADTEN 0n113	OCR1D61 compare match interrupt (ATU4 timer D subblock 6 channel 1 compare match A interrupt) 0: Selection of the above signal as a ΔΣADCn start trigger is disabled. 1: Selection of the above signal as a ΔΣADCn start trigger is enabled.

Table 25.17 PIC2DSADTEN0n1 Register Contents (2/2)

Bit Position	Bit Name	Function
12	PIC2DSADTEN 0n112	OCR1D60 compare match interrupt (ATU4 timer D subblock 6 channel 0 compare match A interrupt) 0: Selection of the above signal as a ΔΣADCn start trigger is disabled. 1: Selection of the above signal as a ΔΣADCn start trigger is enabled.
11	PIC2DSADTEN 0n111	OCR1D43 compare match interrupt (ATU4 timer D subblock 4 channel 3 compare match A interrupt) 0: Selection of the above signal as a ΔΣADCn start trigger is disabled. 1: Selection of the above signal as a ΔΣADCn start trigger is enabled.
10	PIC2DSADTEN 0n110	OCR1D42 compare match interrupt (ATU4 timer D subblock 4 channel 2 compare match A interrupt) 0: Selection of the above signal as a ΔΣADCn start trigger is disabled. 1: Selection of the above signal as a ΔΣADCn start trigger is enabled.
9	PIC2DSADTEN 0n19	OCR1D41 compare match interrupt (ATU4 timer D subblock 4 channel 1 compare match A interrupt) 0: Selection of the above signal as a ΔΣADCn start trigger is disabled. 1: Selection of the above signal as a ΔΣADCn start trigger is enabled.
8	PIC2DSADTEN 0n18	OCR1D40 compare match interrupt (ATU4 timer D subblock 4 channel 0 compare match A interrupt) 0: Selection of the above signal as a ΔΣADCn start trigger is disabled. 1: Selection of the above signal as a ΔΣADCn start trigger is enabled.
7	PIC2DSADTEN 0n17	OCR1D23 compare match interrupt (ATU4 timer D subblock 2 channel 3 compare match A interrupt) 0: Selection of the above signal as a ΔΣADCn start trigger is disabled. 1: Selection of the above signal as a ΔΣADCn start trigger is enabled.
6	PIC2DSADTEN 0n16	OCR1D22 compare match interrupt (ATU4 timer D subblock 2 channel 2 compare match A interrupt) 0: Selection of the above signal as a ΔΣADCn start trigger is disabled. 1: Selection of the above signal as a ΔΣADCn start trigger is enabled.
5	PIC2DSADTEN 0n15	OCR1D21 compare match interrupt (ATU4 timer D subblock 2 channel 1 compare match A interrupt) 0: Selection of the above signal as a ΔΣADCn start trigger is disabled. 1: Selection of the above signal as a ΔΣADCn start trigger is enabled.
4	PIC2DSADTEN 0n14	OCR1D20 compare match interrupt (ATU4 timer D subblock 2 channel 0 compare match A interrupt) 0: Selection of the above signal as a ΔΣADCn start trigger is disabled. 1: Selection of the above signal as a ΔΣADCn start trigger is enabled.
3	PIC2DSADTEN 0n13	OCR1D03 compare match interrupt (ATU4 timer D subblock 0 channel 3 compare match A interrupt) 0: Selection of the above signal as a ΔΣADCn start trigger is disabled. 1: Selection of the above signal as a ΔΣADCn start trigger is enabled.
2	PIC2DSADTEN 0n12	OCR1D02 compare match interrupt (ATU4 timer D subblock 0 channel 2 compare match A interrupt) 0: Selection of the above signal as a ΔΣADCn start trigger is disabled. 1: Selection of the above signal as a ΔΣADCn start trigger is enabled.
1	PIC2DSADTEN 0n11	OCR1D01 compare match interrupt (ATU4 timer D subblock 0 channel 1 compare match A interrupt) 0: Selection of the above signal as a ΔΣADCn start trigger is disabled. 1: Selection of the above signal as a ΔΣADCn start trigger is enabled.
0	PIC2DSADTEN 0n10	OCR1D00 compare match interrupt (ATU4 timer D subblock 0 channel 0 compare match A interrupt) 0: Selection of the above signal as a ΔΣADCn start trigger is disabled. 1: Selection of the above signal as a ΔΣADCn start trigger is enabled.

25.2.3.8 PIC2DSADTEN0n2 — DSADC Start Trigger Output Control Register n2 (n = 0 to 7)

DSADC start trigger output control register n2 enables a trigger source from ATU timer D compare match B to be selected as a trigger for starting ΔΣADC.

Access: This register can be read/written in 32-bit units.

Address: <Base> + 108_H (n = 0), <Base> + 120_H (n = 1), <Base> + 138_H (n = 2), <Base> + 150_H (n = 3)
<Base> + 168_H (n = 4), <Base> + 180_H (n = 5), <Base> + 198_H (n = 6), <Base> + $1B0_H$ (n = 7)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	PIC2DSADTEN 0n219	PIC2DSADTEN 0n218	PIC2DSADTEN 0n217	PIC2DSADTEN 0n216
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PIC2DSADTEN 0n215	PIC2DSADTEN 0n214	PIC2DSADTEN 0n213	PIC2DSADTEN 0n212	PIC2DSADTEN 0n211	PIC2DSADTEN 0n210	PIC2DSADTEN 0n29	PIC2DSADTEN 0n28	PIC2DSADTEN 0n27	PIC2DSADTEN 0n26	PIC2DSADTEN 0n25	PIC2DSADTEN 0n24	PIC2DSADTEN 0n23	PIC2DSADTEN 0n22	PIC2DSADTEN 0n21	PIC2DSADTEN 0n20
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.18 PIC2DSADTEN0n2 Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 20	—	Reserved When writing to these bits, write 0.
19	PIC2DSADTEN 0n219	OCR2D93 compare match interrupt (ATU4 timer D subblock 9 channel 3 compare match B interrupt) 0: Selection of the above signal as a ΔΣADCn start trigger is disabled. 1: Selection of the above signal as a ΔΣADCn start trigger is enabled.
18	PIC2DSADTEN 0n218	OCR2D92 compare match interrupt (ATU4 timer D subblock 9 channel 2 compare match B interrupt) 0: Selection of the above signal as a ΔΣADCn start trigger is disabled. 1: Selection of the above signal as a ΔΣADCn start trigger is enabled.
17	PIC2DSADTEN 0n217	OCR2D91 compare match interrupt (ATU4 timer D subblock 9 channel 1 compare match B interrupt) 0: Selection of the above signal as a ΔΣADCn start trigger is disabled. 1: Selection of the above signal as a ΔΣADCn start trigger is enabled.
16	PIC2DSADTEN 0n216	OCR2D90 compare match interrupt (ATU4 timer D subblock 9 channel 0 compare match B interrupt) 0: Selection of the above signal as a ΔΣADCn start trigger is disabled. 1: Selection of the above signal as a ΔΣADCn start trigger is enabled.
15	PIC2DSADTEN 0n215	OCR2D63 compare match interrupt (ATU4 timer D subblock 6 channel 3 compare match B interrupt) 0: Selection of the above signal as a ΔΣADCn start trigger is disabled. 1: Selection of the above signal as a ΔΣADCn start trigger is enabled.
14	PIC2DSADTEN 0n214	OCR2D62 compare match interrupt (ATU4 timer D subblock 6 channel 2 compare match B interrupt) 0: Selection of the above signal as a ΔΣADCn start trigger is disabled. 1: Selection of the above signal as a ΔΣADCn start trigger is enabled.
13	PIC2DSADTEN 0n213	OCR2D61 compare match interrupt (ATU4 timer D subblock 6 channel 1 compare match B interrupt) 0: Selection of the above signal as a ΔΣADCn start trigger is disabled. 1: Selection of the above signal as a ΔΣADCn start trigger is enabled.
12	PIC2DSADTEN 0n212	OCR2D60 compare match interrupt (ATU4 timer D subblock 6 channel 0 compare match B interrupt) 0: Selection of the above signal as a ΔΣADCn start trigger is disabled. 1: Selection of the above signal as a ΔΣADCn start trigger is enabled.

Table 25.18 PIC2DSADTEN0n2 Register Contents (2/2)

Bit Position	Bit Name	Function
11	PIC2DSADTEN 0n211	OCR2D43 compare match interrupt (ATU4 timer D subblock 4 channel 3 compare match B interrupt) 0: Selection of the above signal as a ΔΣADCn start trigger is disabled. 1: Selection of the above signal as a ΔΣADCn start trigger is enabled.
10	PIC2DSADTEN 0n210	OCR2D42 compare match interrupt (ATU4 timer D subblock 4 channel 2 compare match B interrupt) 0: Selection of the above signal as a ΔΣADCn start trigger is disabled. 1: Selection of the above signal as a ΔΣADCn start trigger is enabled.
9	PIC2DSADTEN 0n29	OCR2D41 compare match interrupt (ATU4 timer D subblock 4 channel 1 compare match B interrupt) 0: Selection of the above signal as a ΔΣADCn start trigger is disabled. 1: Selection of the above signal as a ΔΣADCn start trigger is enabled.
8	PIC2DSADTEN 0n28	OCR2D40 compare match interrupt (ATU4 timer D subblock 4 channel 0 compare match B interrupt) 0: Selection of the above signal as a ΔΣADCn start trigger is disabled. 1: Selection of the above signal as a ΔΣADCn start trigger is enabled.
7	PIC2DSADTEN 0n27	OCR2D23 compare match interrupt (ATU4 timer D subblock 2 channel 3 compare match B interrupt) 0: Selection of the above signal as a ΔΣADCn start trigger is disabled. 1: Selection of the above signal as a ΔΣADCn start trigger is enabled.
6	PIC2DSADTEN 0n26	OCR2D22 compare match interrupt (ATU4 timer D subblock 2 channel 2 compare match B interrupt) 0: Selection of the above signal as a ΔΣADCn start trigger is disabled. 1: Selection of the above signal as a ΔΣADCn start trigger is enabled.
5	PIC2DSADTEN 0n25	OCR2D21 compare match interrupt (ATU4 timer D subblock 2 channel 1 compare match B interrupt) 0: Selection of the above signal as a ΔΣADCn start trigger is disabled. 1: Selection of the above signal as a ΔΣADCn start trigger is enabled.
4	PIC2DSADTEN 0n24	OCR2D20 compare match interrupt (ATU4 timer D subblock 2 channel 0 compare match B interrupt) 0: Selection of the above signal as a ΔΣADCn start trigger is disabled. 1: Selection of the above signal as a ΔΣADCn start trigger is enabled.
3	PIC2DSADTEN 0n23	OCR2D03 compare match interrupt (ATU4 timer D subblock 0 channel 3 compare match B interrupt) 0: Selection of the above signal as a ΔΣADCn start trigger is disabled. 1: Selection of the above signal as a ΔΣADCn start trigger is enabled.
2	PIC2DSADTEN 0n22	OCR2D02 compare match interrupt (ATU4 timer D subblock 0 channel 2 compare match B interrupt) 0: Selection of the above signal as a ΔΣADCn start trigger is disabled. 1: Selection of the above signal as a ΔΣADCn start trigger is enabled.
1	PIC2DSADTEN 0n21	OCR2D01 compare match interrupt (ATU4 timer D subblock 0 channel 1 compare match B interrupt) 0: Selection of the above signal as a ΔΣADCn start trigger is disabled. 1: Selection of the above signal as a ΔΣADCn start trigger is enabled.
0	PIC2DSADTEN 0n20	OCR2D00 compare match interrupt (ATU4 timer D subblock 0 channel 0 compare match B interrupt) 0: Selection of the above signal as a ΔΣADCn start trigger is disabled. 1: Selection of the above signal as a ΔΣADCn start trigger is enabled.

25.2.3.9 PIC2DSADTEN1n0 — DSADC Stop Trigger Output Control Register n0 (n = 0 to 7)

DSADC stop trigger output control register n0 enables a trigger source from ATU timer G compare match to be selected as a trigger for stopping ΔΣADC.

Access: This register can be read/written in 16-bit units.

Address: <Base> + $10C_H$ (n = 0), <Base> + 124_H (n = 1), <Base> + $13C_H$ (n = 2), <Base> + 154_H (n = 3), <Base> + $16C_H$ (n = 4), <Base> + 184_H (n = 5), <Base> + $19C_H$ (n = 6), <Base> + $1B4_H$ (n = 7)

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PIC2DSADTEN1n015	PIC2DSADTEN1n014	PIC2DSADTEN1n013	PIC2DSADTEN1n012	PIC2DSADTEN1n011	PIC2DSADTEN1n010	PIC2DSADTEN1n09	PIC2DSADTEN1n08	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R

Table 25.19 PIC2DSADTEN1n0 Register Contents

Bit Position	Bit Name	Function
15	PIC2DSADTEN1n115	OCRG7 compare match interrupt (ATU4 timer G subblock 7) 0: Selection of the above signal as a ΔΣADCn end trigger is disabled. 1: Selection of the above signal as a ΔΣADCn end trigger is enabled.
14	PIC2DSADTEN1n114	OCRG6 compare match interrupt (ATU4 timer G subblock 6) 0: Selection of the above signal as a ΔΣADCn end trigger is disabled. 1: Selection of the above signal as a ΔΣADCn end trigger is enabled.
13	PIC2DSADTEN1n113	OCRG5 compare match interrupt (ATU4 timer G subblock 5) 0: Selection of the above signal as a ΔΣADCn end trigger is disabled. 1: Selection of the above signal as a ΔΣADCn end trigger is enabled.
12	PIC2DSADTEN1n112	OCRG4 compare match interrupt (ATU4 timer G subblock 4) 0: Selection of the above signal as a ΔΣADCn end trigger is disabled. 1: Selection of the above signal as a ΔΣADCn end trigger is enabled.
11	PIC2DSADTEN1n111	OCRG3 compare match interrupt (ATU4 timer G subblock 3) 0: Selection of the above signal as a ΔΣADCn end trigger is disabled. 1: Selection of the above signal as a ΔΣADCn end trigger is enabled.
10	PIC2DSADTEN1n110	OCRG2 compare match interrupt (ATU4 timer G subblock 2) 0: Selection of the above signal as a ΔΣADCn end trigger is disabled. 1: Selection of the above signal as a ΔΣADCn end trigger is enabled.
9	PIC2DSADTEN1n109	OCRG1 compare match interrupt (ATU4 timer G subblock 1) 0: Selection of the above signal as a ΔΣADCn end trigger is disabled. 1: Selection of the above signal as a ΔΣADCn end trigger is enabled.
8	PIC2DSADTEN1n108	OCRG0 compare match interrupt (ATU4 timer G subblock 0) 0: Selection of the above signal as a ΔΣADCn end trigger is disabled. 1: Selection of the above signal as a ΔΣADCn end trigger is enabled.
7 to 0	—	Reserved When writing to these bits, write 0.

25.2.3.10 PIC2DSADTEN1n1 — DSADC Stop Trigger Output Control Register n1 (n = 0 to 7)

DSADC stop trigger output control register n1 enables a trigger source from ATU timer D compare match A to be selected as a trigger for stopping ΔΣADC.

Access: This register can be read/written in 32-bit units.

Address: <Base> + 110_H (n = 0), <Base> + 128_H (n = 1), <Base> + 140_H (n = 2), <Base> + 158_H (n = 3), <Base> + 170_H (n = 4), <Base> + 188_H (n = 5), <Base> + $1A0_H$ (n = 6), <Base> + $1B8_H$ (n = 7)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	PIC2DSADTEN1n119	PIC2DSADTEN1n118	PIC2DSADTEN1n117	PIC2DSADTEN1n116
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PIC2DSADTEN1n115	PIC2DSADTEN1n114	PIC2DSADTEN1n113	PIC2DSADTEN1n112	PIC2DSADTEN1n111	PIC2DSADTEN1n110	PIC2DSADTEN1n19	PIC2DSADTEN1n18	PIC2DSADTEN1n17	PIC2DSADTEN1n16	PIC2DSADTEN1n15	PIC2DSADTEN1n14	PIC2DSADTEN1n13	PIC2DSADTEN1n12	PIC2DSADTEN1n11	PIC2DSADTEN1n10
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.20 PIC2DSADTEN1n1 Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 20	—	Reserved When writing to these bits, write 0.
19	PIC2DSADTEN1n119	OCR1D93 compare match interrupt (ATU4 timer D subblock 9 channel 3 compare match A interrupt) 0: Selection of the above signal as a ΔΣADCn end trigger is disabled. 1: Selection of the above signal as a ΔΣADCn end trigger is enabled.
18	PIC2DSADTEN1n118	OCR1D92 compare match interrupt (ATU4 timer D subblock 9 channel 2 compare match A interrupt) 0: Selection of the above signal as a ΔΣADCn end trigger is disabled. 1: Selection of the above signal as a ΔΣADCn end trigger is enabled.
17	PIC2DSADTEN1n117	OCR1D91 compare match interrupt (ATU4 timer D subblock 9 channel 1 compare match A interrupt) 0: Selection of the above signal as a ΔΣADCn end trigger is disabled. 1: Selection of the above signal as a ΔΣADCn end trigger is enabled.
16	PIC2DSADTEN1n116	OCR1D90 compare match interrupt (ATU4 timer D subblock 9 channel 0 compare match A interrupt) 0: Selection of the above signal as a ΔΣADCn end trigger is disabled. 1: Selection of the above signal as a ΔΣADCn end trigger is enabled.
15	PIC2DSADTEN1n115	OCR1D63 compare match interrupt (ATU4 timer D subblock 6 channel 3 compare match A interrupt) 0: Selection of the above signal as a ΔΣADCn end trigger is disabled. 1: Selection of the above signal as a ΔΣADCn end trigger is enabled.
14	PIC2DSADTEN1n114	OCR1D62 compare match interrupt (ATU4 timer D subblock 6 channel 2 compare match A interrupt) 0: Selection of the above signal as a ΔΣADCn end trigger is disabled. 1: Selection of the above signal as a ΔΣADCn end trigger is enabled.
13	PIC2DSADTEN1n113	OCR1D61 compare match interrupt (ATU4 timer D subblock 6 channel 1 compare match A interrupt) 0: Selection of the above signal as a ΔΣADCn end trigger is disabled. 1: Selection of the above signal as a ΔΣADCn end trigger is enabled.

Table 25.20 PIC2DSADTEN1n1 Register Contents (2/2)

Bit Position	Bit Name	Function
12	PIC2DSADTEN 1n112	OCR1D60 compare match interrupt (ATU4 timer D subblock 6 channel 0 compare match A interrupt) 0: Selection of the above signal as a ΔΣADCn end trigger is disabled. 1: Selection of the above signal as a ΔΣADCn end trigger is enabled.
11	PIC2DSADTEN 1n111	OCR1D43 compare match interrupt (ATU4 timer D subblock 4 channel 3 compare match A interrupt) 0: Selection of the above signal as a ΔΣADCn end trigger is disabled. 1: Selection of the above signal as a ΔΣADCn end trigger is enabled.
10	PIC2DSADTEN 1n110	OCR1D42 compare match interrupt (ATU4 timer D subblock 4 channel 2 compare match A interrupt) 0: Selection of the above signal as a ΔΣADCn end trigger is disabled. 1: Selection of the above signal as a ΔΣADCn end trigger is enabled.
9	PIC2DSADTEN 1n19	OCR1D41 compare match interrupt (ATU4 timer D subblock 4 channel 1 compare match A interrupt) 0: Selection of the above signal as a ΔΣADCn end trigger is disabled. 1: Selection of the above signal as a ΔΣADCn end trigger is enabled.
8	PIC2DSADTEN 1n18	OCR1D40 compare match interrupt (ATU4 timer D subblock 4 channel 0 compare match A interrupt) 0: Selection of the above signal as a ΔΣADCn end trigger is disabled. 1: Selection of the above signal as a ΔΣADCn end trigger is enabled.
7	PIC2DSADTEN 1n17	OCR1D23 compare match interrupt (ATU4 timer D subblock 2 channel 3 compare match A interrupt) 0: Selection of the above signal as a ΔΣADCn end trigger is disabled. 1: Selection of the above signal as a ΔΣADCn end trigger is enabled.
6	PIC2DSADTEN 1n16	OCR1D22 compare match interrupt (ATU4 timer D subblock 2 channel 2 compare match A interrupt) 0: Selection of the above signal as a ΔΣADCn end trigger is disabled. 1: Selection of the above signal as a ΔΣADCn end trigger is enabled.
5	PIC2DSADTEN 1n15	OCR1D21 compare match interrupt (ATU4 timer D subblock 2 channel 1 compare match A interrupt) 0: Selection of the above signal as a ΔΣADCn end trigger is disabled. 1: Selection of the above signal as a ΔΣADCn end trigger is enabled.
4	PIC2DSADTEN 1n14	OCR1D20 compare match interrupt (ATU4 timer D subblock 2 channel 0 compare match A interrupt) 0: Selection of the above signal as a ΔΣADCn end trigger is disabled. 1: Selection of the above signal as a ΔΣADCn end trigger is enabled.
3	PIC2DSADTEN 1n13	OCR1D03 compare match interrupt (ATU4 timer D subblock 0 channel 3 compare match A interrupt) 0: Selection of the above signal as a ΔΣADCn end trigger is disabled. 1: Selection of the above signal as a ΔΣADCn end trigger is enabled.
2	PIC2DSADTEN 1n12	OCR1D02 compare match interrupt (ATU4 timer D subblock 0 channel 2 compare match A interrupt) 0: Selection of the above signal as a ΔΣADCn end trigger is disabled. 1: Selection of the above signal as a ΔΣADCn end trigger is enabled.
1	PIC2DSADTEN 1n11	OCR1D01 compare match interrupt (ATU4 timer D subblock 0 channel 1 compare match A interrupt) 0: Selection of the above signal as a ΔΣADCn end trigger is disabled. 1: Selection of the above signal as a ΔΣADCn end trigger is enabled.
0	PIC2DSADTEN 1n10	OCR1D00 compare match interrupt (ATU4 timer D subblock 0 channel 0 compare match A interrupt) 0: Selection of the above signal as a ΔΣADCn end trigger is disabled. 1: Selection of the above signal as a ΔΣADCn end trigger is enabled.

25.2.3.11 PIC2DSADTEN1n2 — DSADC Stop Trigger Output Control Register n2 (n = 0 to 7)

DSADC stop trigger output control register n2 enables a trigger source from ATU timer D compare match B to be selected as a trigger for stopping ΔΣADC.

Access: This register can be read/written in 32-bit units.

Address: <Base> + 114_H (n = 0), <Base> + $12C_H$ (n = 1), <Base> + 144_H (n = 2), <Base> + $15C_H$ (n = 3)
<Base> + 174_H (n = 4), <Base> + $18C_H$ (n = 5), <Base> + $1A4_H$ (n = 6), <Base> + $1BC_H$ (n = 7)

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	PIC2DSADTEN1n219	PIC2DSADTEN1n218	PIC2DSADTEN1n217	PIC2DSADTEN1n216
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PIC2DSADTEN1n215	PIC2DSADTEN1n214	PIC2DSADTEN1n213	PIC2DSADTEN1n212	PIC2DSADTEN1n211	PIC2DSADTEN1n210	PIC2DSADTEN1n29	PIC2DSADTEN1n28	PIC2DSADTEN1n27	PIC2DSADTEN1n26	PIC2DSADTEN1n25	PIC2DSADTEN1n24	PIC2DSADTEN1n23	PIC2DSADTEN1n22	PIC2DSADTEN1n21	PIC2DSADTEN1n20
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.21 PIC2DSADTEN1n2 Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 20	—	Reserved When writing to these bits, write 0.
19	PIC2DSADTEN1n219	OCR2D93 compare match interrupt (ATU4 timer D subblock 9 channel 3 compare match B interrupt) 0: Selection of the above signal as a ΔΣADCn end trigger is disabled. 1: Selection of the above signal as a ΔΣADCn end trigger is enabled.
18	PIC2DSADTEN1n218	OCR2D92 compare match interrupt (ATU4 timer D subblock 9 channel 2 compare match B interrupt) 0: Selection of the above signal as a ΔΣADCn end trigger is disabled. 1: Selection of the above signal as a ΔΣADCn end trigger is enabled.
17	PIC2DSADTEN1n217	OCR2D91 compare match interrupt (ATU4 timer D subblock 9 channel 1 compare match B interrupt) 0: Selection of the above signal as a ΔΣADCn end trigger is disabled. 1: Selection of the above signal as a ΔΣADCn end trigger is enabled.
16	PIC2DSADTEN1n216	OCR2D90 compare match interrupt (ATU4 timer D subblock 9 channel 0 compare match B interrupt) 0: Selection of the above signal as a ΔΣADCn end trigger is disabled. 1: Selection of the above signal as a ΔΣADCn end trigger is enabled.
15	PIC2DSADTEN1n215	OCR2D63 compare match interrupt (ATU4 timer D subblock 6 channel 3 compare match B interrupt) 0: Selection of the above signal as a ΔΣADCn end trigger is disabled. 1: Selection of the above signal as a ΔΣADCn end trigger is enabled.
14	PIC2DSADTEN1n214	OCR2D62 compare match interrupt (ATU4 timer D subblock 6 channel 2 compare match B interrupt) 0: Selection of the above signal as a ΔΣADCn end trigger is disabled. 1: Selection of the above signal as a ΔΣADCn end trigger is enabled.
13	PIC2DSADTEN1n213	OCR2D61 compare match interrupt (ATU4 timer D subblock 6 channel 1 compare match B interrupt) 0: Selection of the above signal as a ΔΣADCn end trigger is disabled. 1: Selection of the above signal as a ΔΣADCn end trigger is enabled.
12	PIC2DSADTEN1n212	OCR2D60 compare match interrupt (ATU4 timer D subblock 6 channel 0 compare match B interrupt) 0: Selection of the above signal as a ΔΣADCn end trigger is disabled. 1: Selection of the above signal as a ΔΣADCn end trigger is enabled.

Table 25.21 PIC2DSADTEN1n2 Register Contents (2/2)

Bit Position	Bit Name	Function
11	PIC2DSADTEN 1n211	OCR2D43 compare match interrupt (ATU4 timer D subblock 4 channel 3 compare match B interrupt) 0: Selection of the above signal as a ΔΣADCn end trigger is disabled. 1: Selection of the above signal as a ΔΣADCn end trigger is enabled.
10	PIC2DSADTEN 1n210	OCR2D42 compare match interrupt (ATU4 timer D subblock 4 channel 2 compare match B interrupt) 0: Selection of the above signal as a ΔΣADCn end trigger is disabled. 1: Selection of the above signal as a ΔΣADCn end trigger is enabled.
9	PIC2DSADTEN 1n29	OCR2D41 compare match interrupt (ATU4 timer D subblock 4 channel 1 compare match B interrupt) 0: Selection of the above signal as a ΔΣADCn end trigger is disabled. 1: Selection of the above signal as a ΔΣADCn end trigger is enabled.
8	PIC2DSADTEN 1n28	OCR2D40 compare match interrupt (ATU4 timer D subblock 4 channel 0 compare match B interrupt) 0: Selection of the above signal as a ΔΣADCn end trigger is disabled. 1: Selection of the above signal as a ΔΣADCn end trigger is enabled.
7	PIC2DSADTEN 1n27	OCR2D23 compare match interrupt (ATU4 timer D subblock 2 channel 3 compare match B interrupt) 0: Selection of the above signal as a ΔΣADCn end trigger is disabled. 1: Selection of the above signal as a ΔΣADCn end trigger is enabled.
6	PIC2DSADTEN 1n26	OCR2D22 compare match interrupt (ATU4 timer D subblock 2 channel 2 compare match B interrupt) 0: Selection of the above signal as a ΔΣADCn end trigger is disabled. 1: Selection of the above signal as a ΔΣADCn end trigger is enabled.
5	PIC2DSADTEN 1n25	OCR2D21 compare match interrupt (ATU4 timer D subblock 2 channel 1 compare match B interrupt) 0: Selection of the above signal as a ΔΣADCn end trigger is disabled. 1: Selection of the above signal as a ΔΣADCn end trigger is enabled.
4	PIC2DSADTEN 1n24	OCR2D20 compare match interrupt (ATU4 timer D subblock 2 channel 0 compare match B interrupt) 0: Selection of the above signal as a ΔΣADCn end trigger is disabled. 1: Selection of the above signal as a ΔΣADCn end trigger is enabled.
3	PIC2DSADTEN 1n23	OCR2D03 compare match interrupt (ATU4 timer D subblock 0 channel 3 compare match B interrupt) 0: Selection of the above signal as a ΔΣADCn end trigger is disabled. 1: Selection of the above signal as a ΔΣADCn end trigger is enabled.
2	PIC2DSADTEN 1n22	OCR2D02 compare match interrupt (ATU4 timer D subblock 0 channel 2 compare match B interrupt) 0: Selection of the above signal as a ΔΣADCn end trigger is disabled. 1: Selection of the above signal as a ΔΣADCn end trigger is enabled.
1	PIC2DSADTEN 1n21	OCR2D01 compare match interrupt (ATU4 timer D subblock 0 channel 1 compare match B interrupt) 0: Selection of the above signal as a ΔΣADCn end trigger is disabled. 1: Selection of the above signal as a ΔΣADCn end trigger is enabled.
0	PIC2DSADTEN 1n20	OCR2D00 compare match interrupt (ATU4 timer D subblock 0 channel 0 compare match B interrupt) 0: Selection of the above signal as a ΔΣADCn end trigger is disabled. 1: Selection of the above signal as a ΔΣADCn end trigger is enabled.

25.2.3.12 PIC2DSADCATSEL0 — DSADC Trigger Selection Control Register 0

DSADC trigger selection control register 0 selects a trigger for starting DSADCj (j = 0 to 7).

Access: This register can be read/written in 16-bit units.

Address: <Base> + $1C0_H$

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DSADCATSEL015	DSADCATSEL014	DSADCATSEL013	DSADCATSEL012	DSADCATSEL011	DSADCATSEL010	DSADCATSEL009	DSADCATSEL008	DSADCATSEL007	DSADCATSEL006	DSADCATSEL005	DSADCATSEL004	DSADCATSEL003	DSADCATSEL002	DSADCATSEL001	DSADCATSEL000
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.22 PIC2DSADCATSEL0 Register Contents (1/2)

Bit Position	Bit Name	Function
15	DSADCATSEL015	Selects the DSADTRG7 pin as a trigger source for starting DSADC7. 0: DSADTRG7 is not selected. 1: DSADTRG7 is selected.
14	DSADCATSEL014	Selects the trigger selected by DSADC start trigger output control register 7j (j = 0 to 2) as a trigger source for starting DSADC7. 0: Does not select the trigger selected by register PIC2DSADTEN07j. 1: Selects the trigger selected by register PIC2DSADTEN07j.
13	DSADCATSEL013	Selects the DSADTRG6 pin as a trigger source for starting DSADC6. 0: DSADTRG6 is not selected. 1: DSADTRG6 is selected.
12	DSADCATSEL012	Selects the trigger selected by DSADC start trigger output control register 6j (j = 0 to 2) as a trigger source for starting DSADC6. 0: Does not select the trigger selected by register PIC2DSADTEN06j. 1: Selects the trigger selected by register PIC2DSADTEN06j.
11	DSADCATSEL011	Selects the DSADTRG5 pin as a trigger source for starting DSADC5. 0: DSADTRG5 is not selected. 1: DSADTRG5 is selected.
10	DSADCATSEL010	Selects the trigger selected by DSADC start trigger output control register 5j (j = 0 to 2) as a trigger source for starting DSADC5. 0: Does not select the trigger selected by register PIC2DSADTEN05j. 1: Selects the trigger selected by register PIC2DSADTEN05j.
9	DSADCATSEL009	Selects the DSADTRG4 pin as a trigger source for starting DSADC4. 0: DSADTRG4 is not selected. 1: DSADTRG4 is selected.
8	DSADCATSEL008	Selects the trigger selected by DSADC start trigger output control register 4j (j = 0 to 2) as a trigger source for starting DSADC4. 0: Does not select the trigger selected by register PIC2DSADTEN04j. 1: Selects the trigger selected by register PIC2DSADTEN04j.
7	DSADCATSEL007	Selects the DSADTRG3 pin as a trigger source for starting DSADC3. 0: DSADTRG3 is not selected. 1: DSADTRG3 is selected.
6	DSADCATSEL006	Selects the trigger selected by DSADC start trigger output control register 3j (j = 0 to 2) as a trigger source for starting DSADC3. 0: Does not select the trigger selected by register PIC2DSADTEN03j. 1: Selects the trigger selected by register PIC2DSADTEN03j.
5	DSADCATSEL005	Selects the DSADTRG2 pin as a trigger source for starting DSADC2. 0: DSADTRG2 is not selected. 1: DSADTRG2 is selected.
4	DSADCATSEL004	Selects the trigger selected by DSADC start trigger output control register 2j (j = 0 to 2) as a trigger source for starting DSADC2. 0: Does not select the trigger selected by register PIC2DSADTEN02j. 1: Selects the trigger selected by register PIC2DSADTEN02j.

Table 25.22 PIC2DSADCATSEL0 Register Contents (2/2)

Bit Position	Bit Name	Function
3	DSADCATSEL003	Selects the DSADTRG1 pin as a trigger source for starting DSADC1. 0: DSADTRG1 is not selected. 1: DSADTRG1 is selected.
2	DSADCATSEL002	Selects the trigger selected by DSADC start trigger output control register 1j (j = 0 to 2) as a trigger source for starting DSADC1. 0: Does not select the trigger selected by register PIC2DSADTEN01j. 1: Selects the trigger selected by register PIC2DSADTEN01j.
1	DSADCATSEL001	Selects the DSADTRG0 pin as a trigger source for starting DSADC0. 0: DSADTRG0 is not selected. 1: DSADTRG0 is selected.
0	DSADCATSEL000	Selects the trigger selected by DSADC start trigger output control register 0j (j = 0 to 2) as a trigger source for starting DSADC0. 0: Does not select the trigger selected by register PIC2DSADTEN00j. 1: Selects the trigger selected by register PIC2DSADTEN00j.

25.2.3.13 PIC2DSADCATSEL1 — DSADC Trigger Selection Control Register 1

DSADC trigger selection control register 1 selects a trigger for stopping DSADCj (j = 0 to 7).

Access: This register can be read/written in 16-bit units.

Address: <Base> + $1C4_H$

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DSADCATSEL115	DSADCATSEL114	DSADCATSEL113	DSADCATSEL112	DSADCATSEL111	DSADCATSEL110	DSADCATSEL109	DSADCATSEL108	DSADCATSEL107	DSADCATSEL106	DSADCATSEL105	DSADCATSEL104	DSADCATSEL103	DSADCATSEL102	DSADCATSEL101	DSADCATSEL100
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 25.23 PIC2DSADCATSEL1 Register Contents (1/2)

Bit Position	Bit Name	Function
15	DSADCATSEL115	Selects the DSADTRG7 pin as a trigger source for stopping DSADC7. 0: DSADTRG7 is not selected. 1: DSADTRG7 is selected.
14	DSADCATSEL114	Selects the trigger selected by DSADC stop trigger output control register 7j (j = 0 to 2) as a trigger source for stopping DSADC7. 0: Does not select the trigger selected by register PIC2DSADTEN07j. 1: Selects the trigger selected by register PIC2DSADTEN07j.
13	DSADCATSEL113	Selects the DSADTRG6 pin as a trigger source for stopping DSADC6. 0: DSADTRG6 is not selected. 1: DSADTRG6 is selected.
12	DSADCATSEL112	Selects the trigger selected by DSADC stop trigger output control register 6j (j = 0 to 2) as a trigger source for stopping DSADC6. 0: Does not select the trigger selected by register PIC2DSADTEN06j. 1: Selects the trigger selected by register PIC2DSADTEN06j.
11	DSADCATSEL111	Selects the DSADTRG5 pin as a trigger source for stopping DSADC5. 0: DSADTRG5 is not selected. 1: DSADTRG5 is selected.
10	DSADCATSEL110	Selects the trigger selected by DSADC stop trigger output control register 5j (j = 0 to 2) as a trigger source for stopping DSADC5. 0: Does not select the trigger selected by register PIC2DSADTEN15j. 1: Selects the trigger selected by register PIC2DSADTEN15j.
9	DSADCATSEL109	Selects the DSADTRG4 pin as a trigger source for stopping DSADC4. 0: DSADTRG4 is not selected. 1: DSADTRG4 is selected.
8	DSADCATSEL108	Selects the trigger selected by DSADC stop trigger output control register 4j (j = 0 to 2) as a trigger source for stopping DSADC4. 0: Does not select the trigger selected by register PIC2DSADTEN14j. 1: Selects the trigger selected by register PIC2DSADTEN14j.
7	DSADCATSEL107	Selects the DSADTRG3 pin as a trigger source for stopping DSADC3. 0: DSADTRG3 is not selected. 1: DSADTRG3 is selected.
6	DSADCATSEL106	Selects the trigger selected by DSADC stop trigger output control register 3j (j = 0 to 2) as a trigger source for stopping DSADC3. 0: Does not select the trigger selected by register PIC2DSADTEN13j. 1: Selects the trigger selected by register PIC2DSADTEN13j.
5	DSADCATSEL105	Selects the DSADTRG2 pin as a trigger source for stopping DSADC2. 0: DSADTRG2 is not selected. 1: DSADTRG2 is selected.
4	DSADCATSEL104	Selects the trigger selected by DSADC stop trigger output control register 2j (j = 0 to 2) as a trigger source for stopping DSADC2. 0: Does not select the trigger selected by register PIC2DSADTEN12j. 1: Selects the trigger selected by register PIC2DSADTEN12j.

Table 25.23 PIC2DSADCATSEL1 Register Contents (2/2)

Bit Position	Bit Name	Function
3	DSADCATSEL103	Selects the DSADTRG1 pin as a trigger source for stopping DSADC1. 0: DSADTRG1 is not selected. 1: DSADTRG1 is selected.
2	DSADCATSEL102	Selects the trigger selected by DSADC stop trigger output control register 1j (j = 0 to 2) as a trigger source for stopping DSADC1. 0: Does not select the trigger selected by register PIC2DSADTEN11j. 1: Selects the trigger selected by register PIC2DSADTEN11j.
1	DSADCATSEL101	Selects the DSADTRG0 pin as a trigger source for stopping DSADC0. 0: DSADTRG0 is not selected. 1: DSADTRG0 is selected.
0	DSADCATSEL100	Selects the trigger selected by DSADC stop trigger output control register 0j (j = 0 to 2) as a trigger source for stopping DSADC0. 0: Does not select the trigger selected by register PIC2DSADTEN10j. 1: Selects the trigger selected by register PIC2DSADTEN10j.

25.2.4 AD Trigger Selection Function

25.2.4.1 ADC Trigger Selection Function

Each of ADCB0 and ADCB1 has 5 scan groups, and provides the ADCB hardware triggers corresponding to each of the scan groups.

The internal trigger signals selected by register PIC2ADCBnTSELj from TSG20, TSG21, the ATU, and APA0 are ORed with the external trigger signal from the pin and the result can be input as the ADCB hardware trigger signal for the pertinent scan group.

Furthermore, of the above triggers, the ATU and APA triggers are selected from among internal signals of the respective timers.

NOTE

One input external ADTRG pin is provided as an external trigger pin for each AD1. The same external pin is allocated to each AD scan group.

- ATU and APA Trigger Selection
  ADCB hardware trigger signals are generated for 5 scan groups per AD, a total of 10 triggers, from the ATU sources (16 sources of ATU timer C compare match, 20 sources of ATU timer D compare match A, 20 sources of ATU timer D compare match B, 8 sources of ATU timer G compare match, and 1 source of ATU DMA switch function, which are selected independently for each scan group) and the APA0 interrupt.

25.2.4.2 ΔΣADC Trigger Selection Function

ΔΣADC has 2 types of hardware trigger: start and stop triggers. Start and stop triggers are obtained by ORing external pin inputs DSADTRGn and signals selected by ATU trigger selection function and can be input as the ΔΣADC hardware trigger signal.

- ATU trigger selection
  One ΔΣADC trigger is generated for each AD from the ATU sources (20 sources of ATU timer D compare match A, 20 sources of ATU timer D compare match B, and 8 sources of ATU timer G compare match).

Section 26 A/D Converter (ADCB)

This LSI has two 12-bit successive approximation A/D converter modules (ADC0 and ADC1).

26.1 Features

The ADC has the following features.

- Analog channels and sample-and-hold function
  A/D conversion is available for 24 ADC0 channels and 24 ADC1 channels (48 channels in total).
  Each of ADC0 and ADC1 has an internal sample-and-hold circuit that enables ADC0 and ADC1 to perform A/D conversion independently.
- Advanced A/D converter
  Resolution: 12 bits
  A/D conversion method: Successive approximation
  Conversion speed: 1.0 µs
- Supporting simultaneous track and hold for up to 4 channels
  ADC0 has a track-and-hold circuit for 2 channels and ADC1 has for 2 channels , supporting simultaneous track and hold for up to 4 channels.
- Supporting five scan groups
  Each ADC has five scan groups. Scan settings can be made independently for each scan group.
- Two scan modes
  Each ADC has two scan modes.
  Multicycle scan mode: Specified number of scans are executed.
  Continuous scan mode: Scans are repeatedly executed without limit.
- Virtual channels
  Each ADC has 40 virtual channels. Analog channels for which A/D conversion is to be made and other accompanying information are set for each virtual channel. By sequentially performing the processing for the virtual channels indicated by the start virtual channel pointer and the end virtual channel pointer in each scan group, scans (which can perform A/D conversion for any analog channels in any order) can be executed.
- Extended physical channels
  Each ADC can extend physical channels by using an external analog multiplexer. Furthermore, an external analog multiplexer for the CSIH interface is available when DMA is used. (One channel can be set for each ADC.)
- Interval function
  The ADC can start scan groups in any cycle by using the A/D timer equipped in the scan groups 3 and 4. This enables scans with intervals inserted.
- A/D-converted value adding function
  The ADC performs A/D conversion sequentially twice or four times for a channel, and stores the addition result in the data register. The addition count can be set for each virtual channel.
  The effect of the moving average filter can be gained by using this result. However, this function does not always ensure that A/D conversion accuracy is improved.
  In addition, using an optional function ASF makes the extended summation function available.
  For details, see **Section 26.11, ADC Summation Function (ASF)**.

- Data registers
  Data registers corresponding to virtual channels are provided.
- Start trigger for each scan group
  Hardware triggers and software triggers can start processing of each scan group. Only scan groups 3 and 4 can start processing by an A/D timer trigger.
- Asynchronous/synchronous suspend and resume function
  A processing for a scan group can interrupt an ongoing processing for another scan group. The priority is as follows:
  Low High
  SG0 < SG1 < SG2 < SG3 < SG4 (SG: Scan group)
  If a request for a higher-priority SG is present while a lower-priority SG is being processed, the lower-priority SG is suspended after the ongoing virtual channel processing is stopped (synchronous suspend) or after the ongoing virtual channel processing is immediately stopped (asynchronous suspend), and then the processing for the higher-priority SG is performed. After the processing for the higher-priority SG is completed, the suspended virtual channel processing of the lower-priority SG resumes.
  Also, you can set as follows: when a higher-priority SG interrupts an SG0 processing, asynchronous suspend occurs, but when a higher-priority SG interrupts a lower-priority SG other than SG0, synchronous suspend occurs.
- Entry to the digital filter engine and the ADC summation function
  A/D-converted values can be directly entered into the digital filter engine (DFE) or the ADC summation function (ASF). Whether to enable or disable entry and a tag used to define the target channel for entry can be set for each virtual channel. Entry to the DEF or entry to the ASF can be set for each scan group.
- Supporting scan end interrupt and DMA transfer
  Each scan group can generate an interrupt request to the INTC and activate the DMAC each time a processing for the virtual channel indicated by the end virtual channel pointer ends or a virtual channel ends.
- Supporting interrupt and DMA transfer for an external analog multiplexer
  The ADC can generate an interrupt request to the INTC and activate the DMAC when the specified virtual channel is started. This enables transfer of the MPX value to an external analog multiplexer in cooperation with an I/O port or CSIH.
- An analog conversion voltage settable
  The A0VREFH pin and the A1VREFH pin can be used to set the voltage range for analog conversion.
- A/D conversion monitor output
  The processing timing of a desired virtual channel can be output to the A/D conversion monitor output pin.
- A wide range of safety functions
  The ADC has various safety functions, including self-diagnosis, pin-level self-diagnosis, wiring-break detection, normality check for analog selection, upper-limit/lower-limit check for data registers, parity check for data registers, overwrite check for data registers, and read and clear function for data registers.

26.2 Configuration

Figure 26.1 and **Figure 26.2** show the block diagrams of ADC, and **Table 26.1** and **Table 26.2** show the pin configuration of ADC.

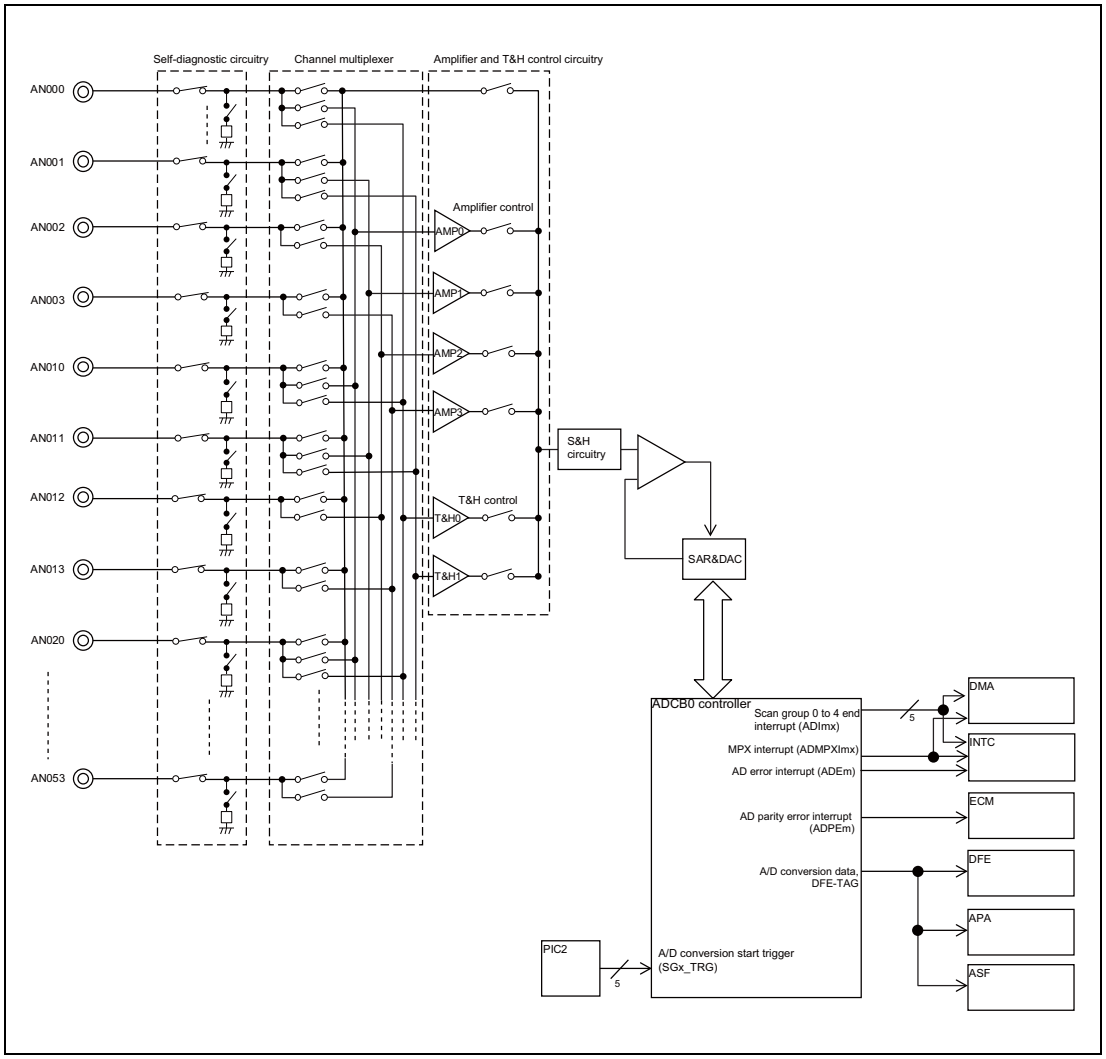


Figure 26.1 ADC0 Block Diagram

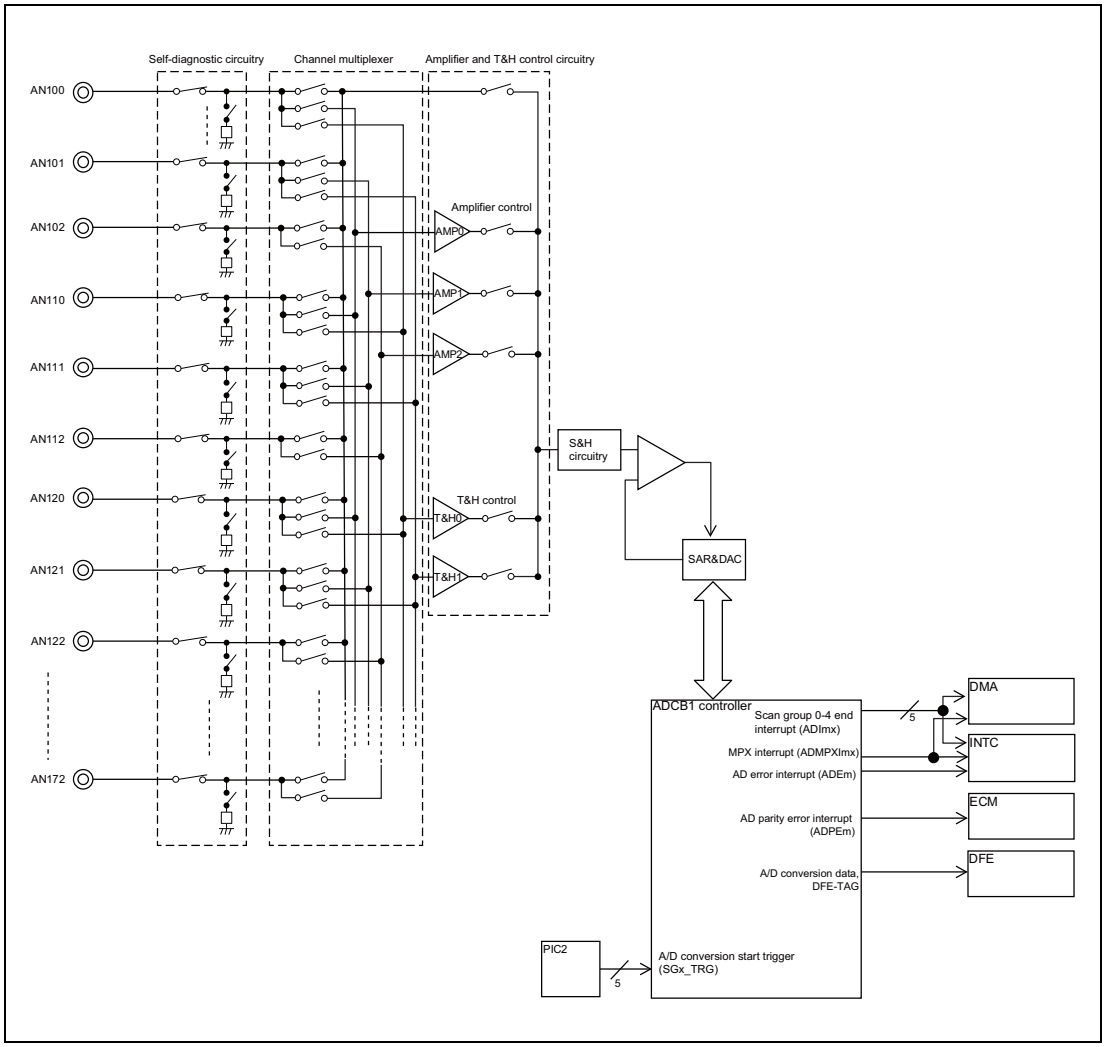


Figure 26.2 ADC1 Block Diagram

Table 26.1 ADC Pin Configuration (1/2)

Pin Name	Module	I/O	Description
A0VCC	ADC0	Input	Analog block power supply pin
A0VSS	ADC0	Input	Analog block ground pin
A1VCC	ADC1	Input	Analog block power supply pin
A1VSS	ADC1	Input	Analog block ground pin
A0VREFH	ADC0	Input	Analog block reference voltage pin
A1VREFH	ADC1	Input	Analog block reference voltage pin
AN000	ADC0	Input	Analog pin 000
AN001	ADC0	Input	Analog pin 001
AN002	ADC0	Input	Analog pin 002
AN003	ADC0	Input	Analog pin 003
AN010	ADC0	Input	Analog pin 010
AN011	ADC0	Input	Analog pin 011
AN012	ADC0	Input	Analog pin 012
AN013	ADC0	Input	Analog pin 013
AN020	ADC0	Input	Analog pin 020
AN021	ADC0	Input	Analog pin 021
AN022	ADC0	Input	Analog pin 022
AN023	ADC0	Input	Analog pin 023
AN030	ADC0	Input	Analog pin 030
AN031	ADC0	Input	Analog pin 031
AN032	ADC0	Input	Analog pin 032
AN033	ADC0	Input	Analog pin 033
AN040	ADC0	Input	Analog pin 040
AN041	ADC0	Input	Analog pin 041
AN042	ADC0	Input	Analog pin 042
AN043	ADC0	Input	Analog pin 043/external analog multiplexer input pin
AN050	ADC0	Input	Analog pin 050
AN051	ADC0	Input	Analog pin 051
AN052	ADC0	Input	Analog pin 052
AN053	ADC0	Input	Analog pin 053
AN100	ADC1	Input	Analog pin 100/external analog multiplexer input pin
AN101	ADC1	Input	Analog pin 101
AN102	ADC1	Input	Analog pin 102
AN110	ADC1	Input	Analog pin 110
AN111	ADC1	Input	Analog pin 111
AN112	ADC1	Input	Analog pin 112
AN120	ADC1	Input	Analog pin 120
AN121	ADC1	Input	Analog pin 121
AN122	ADC1	Input	Analog pin 122
AN130	ADC1	Input	Analog pin 130
AN131	ADC1	Input	Analog pin 131
AN132	ADC1	Input	Analog pin 132
AN140	ADC1	Input	Analog pin 140
AN141	ADC1	Input	Analog pin 141

Table 26.1 ADC Pin Configuration (2/2)

Pin Name	Module	I/O	Description
AN142	ADC1	Input	Analog pin 142
AN150	ADC1	Input	Analog pin 150
AN151	ADC1	Input	Analog pin 151
AN152	ADC1	Input	Analog pin 152
AN160	ADC1	Input	Analog pin 160
AN161	ADC1	Input	Analog pin 161
AN162	ADC1	Input	Analog pin 162
AN170	ADC1	Input	Analog pin 170
AN171	ADC1	Input	Analog pin 171
AN172	ADC1	Input	Analog pin 172
ADTRG0	ADC0	Input	External trigger pin
ADTRG1	ADC1	Input	External trigger pin
ADEND0	ADC0	Output	A/D conversion timing monitor pin
ADEND1	ADC1	Output	A/D conversion timing monitor pin

To ensure reliability of the LSI, the following relationship is required between A0VCC, A1VCC, A0VSS, and A1VSS and VCC and VSS when the ADC is used.

A0VCC = 5V ± 0.5V, A1VCC = 5V ± 0.5V, and A0VSS = A1VSS = VSS

If you do not use the ADC, the following relationship should be satisfied: A0VCC = A1VCC = EVCC, and A0VSS = A1VSS = VSS. The voltage to be applied to analog input pins must be within the following range:

A0VSS ≤ AN0kl (k = 0 to 5, l = 0 to 3) ≤ A0VREFH, and A1VSS ≤ AN1kl (k = 0 to 7, l = 0 to 2) ≤ A1VREFH

Table 26.2 Physical Channel Groups and Physical Sub-Channels

ADC0

Pin Name	Physical Channel Groups	Physical Sub-Channels
AN000	0	0
AN001		1
AN002		2
AN003		3
AN010	1	0
AN011		1
AN012		2
AN013		3
AN020	2	0
AN021		1
AN022		2
AN023		3
AN030	3	0
AN031		1
AN032		2
AN033		3
AN040	4	0
AN041		1
AN042		2
AN043		3
AN050	5	0
AN051		1
AN052		2
AN053		3

ADC1

Pin Name	Physical Channel Groups	Physical Sub-Channels
AN100	0	0
AN101		1
AN102		2
AN110	1	0
AN111		1
AN112		2
AN120	2	0
AN121		1
AN122		2
AN130	3	0
AN131		1
AN132		2
AN140	4	0
AN141		1
AN142		2
AN150	5	0
AN151		1
AN152		2
AN160	6	0
AN161		1
AN162		2
AN170	7	0
AN171		1
AN172		2

26.3 Register Address

Table 26.3 Register Base Address <ADCm_base>

ADCm	<ADCm_base> Address
ADC0	FFF2 0000_H
ADC1	FFF2 1000_H

Note: m = 0, 1

Table 26.4 Register Address List (1/2)

Register Name	Symbol	Value after Reset	Address	Access Size
■ ADC common registers				
A/D synchronization start control register	ADCB0ADSYNSTCR	00_H	<ADC0_base> + 300_H	8
A/D timer synchronization start control register	ADCB0ADTSYNSTCR	00_H	<ADC0_base> + 304_H	8
■ ADC specific registers (virtual channel)				
Virtual channel register n	ADCBmVCRn	0000_H	<ADCm_base> + n × 4_H	8, 16
Data register n	ADCBmDRn	0000_H	<ADCm_base> + 100_H + n × 2_H	16, 32 (n is even number) 16 (n is odd number)
Data supplementary information register n	ADCBmDIRn	0000 0000_H	<ADCm_base> + 200_H + n × 4_H	32
■ ADC specific registers (control)				
A/D halt register	ADCBmADHALTR	00_H	<ADCm_base> + 380_H	8
A/D control register 1	ADCBmADCR1	00_H	<ADCm_base> + 384_H	8
MPX current control register	ADCBmMPXCURCR	00_H	<ADCm_base> + 388_H	8
MPX current register	ADCBmMPXCURR	0000 0000_H	<ADCm_base> + $38C_H$	32
MPX optional wait register	ADCBmMPXOWR	00_H	<ADCm_base> + 390_H	8
MPX command information register	ADCBmMPXCMDR	00_H	<ADCm_base> + 394_H	8
A/D control register 2	ADCBmADCR2	00_H	<ADCm_base> + 398_H	8
DFE/ASF entry scan group enable register	ADCBmDFASENTSGER	0000_H	<ADCm_base> + $39C_H$	16
A/D conversion monitor virtual channel pointer	ADCBmADENDP	00_H	<ADCm_base> + $3A0_H$	8
■ ADC specific registers (safety-related)				
Safety control register	ADCBmSFTCR	00_H	<ADCm_base> + $3C0_H$	8
Pin-Level self-diagnosis control register	ADCBmTDCR	00_H	<ADCm_base> + $3C4_H$	8
Wiring-break detection control register	ADCBmODCR	00_H	<ADCm_base> + $3C8_H$	8
Upper limit/lower limit table register 0	ADCBmULLMTBR0	7FFE 0000_H	<ADCm_base> + $3CC_H$	16, 32
Upper limit/lower limit table register 1	ADCBmULLMTBR1	7FFE 0000_H	<ADCm_base> + $3D0_H$	16, 32
Upper limit/lower limit table register 2	ADCBmULLMTBR2	7FFE 0000_H	<ADCm_base> + $3D4_H$	16, 32
Error clear register	ADCBmECR	00_H	<ADCm_base> + $3D8_H$	8
Upper limit/lower limit error register	ADCBmULER	00_H	<ADCm_base> + $3DC_H$	8
Overwrite error register	ADCBmOWER	00_H	<ADCm_base> + $3E0_H$	8
Parity error register	ADCBmPER	00_H	<ADCm_base> + $3E4_H$	8
ID error register	ADCBmIDER	00_H	<ADCm_base> + $3E8_H$	8
■ Scan group specific registers				
Scan group x start control register	ADCBmSGSTCRx	00_H	<ADCm_base> + x × 80_H + 480_H	8

Table 26.4 Register Address List (2/2)

Register Name	Symbol	Value after Reset	Address	Access Size
A/D timer y start control register	ADCBmADTSTCRy	00_H	<ADCm_base> + y × 80_H + 488_H	8
A/D timer y end control register	ADCBmADTENDCRy	00_H	<ADCm_base> + y × 80_H + $48C_H$	8
Scan group x control register	ADCBmSGCRx	00_H	<ADCm_base> + x × 80_H + 490_H	8
Scan group x start virtual channel pointer	ADCBmSGVCSPx	00_H	<ADCm_base> + x × 80_H + 494_H	8
Scan group x end virtual channel pointer	ADCBmSGVCEPx	00_H	<ADCm_base> + x × 80_H + 498_H	8
Scan group x multicycle register	ADCBmSGMCYCRx	00_H	<ADCm_base> + x × 80_H + $49C_H$	8
Scan group x status register	ADCBmSGSRx	00_H	<ADCm_base> + x × 80_H + $4A4_H$	8
A/D timer initial phase register y	ADCBmADTIPRy	0000 0000_H	<ADCm_base> + y × 80_H + $4A8_H$	32
A/D timer cycle register y	ADCBmADTPRRy	001F $FFFF_H$	<ADCm_base> + y × 80_H + $4AC_H$	32
Scan group x upper limit/lower limit table select register	ADCBmULLMSRx	00_H	<ADCm_base> + x × 80_H + $4B0_H$	8

Note: m = 0, 1; n = 0 to 39; x = 0 to 4; y = 3, 4

26.4 ADC Common Registers

This section describes common registers for ADC0 and ADC1.

26.4.1 ADCB0ADSYNSTCR — A/D Synchronization Start Control Register

ADCB0ADSYNSTCR is an 8-bit write-only register that controls simultaneous start of A/D conversion for scan groups of ADC0 and ADC1. The register bits are always read as 0.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ADSTART
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 26.5 ADCB0ADSYNSTCR Register Contents

Bit Position	Bit Name	Function
7 to 1	—	Reserved These bits are always read as 0. The write value should always be 0.
0	ADSTART	This bit starts A/D conversion for scan groups of ADC0 and ADC1. Start condition for SGx of ADCm: 1 is written to ADSTART when SGACT for SGx of ADCm is 0 and ADSTARTE is 1. A/D conversion is started simultaneously for the scan groups (of ADC0 and ADC1) for which ADSTARTE has been set to 1.

NOTE

m = 0, 1; x = 0 to 4

26.4.2 ADCB0ADTSYNSTCR — A/D Timer Synchronization Start Control Register

ADCB0ADTSYNSTCR is an 8-bit write-only register that controls simultaneous start of counting by A/D timers of ADC0 and ADC1. The register bits are always read as 0.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ADTSTART
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 26.6 ADCB0ADTSYNSTCR Register Contents

Bit Position	Bit Name	Function
7 to 1	—	Reserved These bits are always read as 0. The write value should always be 0.
0	ADTSTART	This bit starts counting by the A/D timers of ADC0 and ADC1. Start condition for SGx of ADCm: 1 is written to ADTSTART when ADTACT for A/D timer y of ADCm is 0 and ADTSTARTE is 1. The A/D timers (of ADC0 and ADC1) for which ADTST has been set to 1 start counting simultaneously.

NOTE

m = 0, 1; x = 0 to 4; y = 3, 4

26.5 ADC Specific Registers (Virtual Channel)

26.5.1 ADCBmVCRn — Virtual Channel Register n

ADCBmVCRn is a 16-bit readable/writable register used for each virtual channel. ADCBmVCRn is initialized to 0000_H at a reset.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CNVCLS[2:0]			DFENT	DFTAG[3:0]				ADIE	—	GCTRL[5:0]					
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 26.7 ADCBmVCRn Register Contents (1/2)

Bit Position	Bit Name	Function
15 to 13	CNVCLS[2:0]	Conversion Type 0_H: Normal A/D conversion 1_H: Hold value A/D conversion 2_H: Execution of the selected T&H (physical channel group selected) 3_H: Self-diagnosis 4_H: Normal A/D conversion in addition mode 5_H: Normal A/D conversion with the MPX 6_H: Normal A/D conversion with the MPX in addition mode Other than above: Setting prohibited
12	DFENT	DFE Entry 0: Entry disabled 1: Entry enabled This bit specifies whether to enable or disable entry to DFE (digital filter engine) or ASF (ADC summation function). Only the scan groups enabled by DFENTSGxE and ASENTSGxE in ADCBmDFASENTSGER are entered. No scan groups are entered for execution of the selected T&H (CNVCLS[2:0] = 2_H) regardless of the DFENT setting.
11 to 8	DFTAG[3:0]	DFE-TAG When entry to DFE is requested, entry is made to the DFE channel for which the same tag with DFTAG[3:0] is set. If this applies to multiple channels, entry is made to multiple channels. When entry to ASF is requested, entry is made to the ASF channel corresponding to DFTAG[3:0]. When entry to ASF is requested, setting of DFTAG[3:0] = B_H to F_H is invalid.
7	ADIE	Virtual Channel End Interrupt Enable 0: ADImx is not output when A/D conversion for virtual channel n ends in SGx 1: ADImx is output when A/D conversion for virtual channel n ends in SGx ADIE in ADCBmSGCRx is independent of ADIE in ADCBmVCRn. For details, see **Section 26.7.16, Scan End Interrupt Request**.
6	—	Reserved This bit is always read as 0. The write value should always be 0.
5 to 0	GCTRL[5:0]	General Control • For normal A/D conversion (CNVCLS[2:0] = 0_H) – GCTRL[4:2]: Physical Channel Group These bits specify a physical channel group. – GCTRL[1:0]: Physical sub-channel These bits specify a physical sub-channel. Always set the other GCTRL bits to 0.

Table 26.7 ADCBmVCRn Register Contents (2/2)

Bit Position	Bit Name	Function
5 to 0	GCTRL[5:0]	General Control (continued) • For hold value A/D conversion (CNVCLS[2:0] = 1_H) – GCTRL[2:0]: Specification of T&H 0_H: Hold value of T&H0 is A/D converted. 1_H: Hold value of T&H1 is A/D converted. Settings other than the above and those described in **Section 26.9.3, Notes when Current is Being Injected**, are prohibited. • For execution of the selected T&H (CNVCLS[2:0] = 2_H) – GCTRL[5:3]: Physical Channel Group These bits specify a physical channel group. – GCTRL[1]: T&H1 Execution 0: Sampling and holding are not executed on T&H1. 1: Sampling and holding of physical sub-channel 1 are executed on T&H1. – GCTRL[0]: T&H0 Execution 0: Sampling and holding are not executed on T&H0. 1: Sampling and holding of physical sub-channel 0 are executed on T&H0. Settings other than the above and those described in **Section 26.9.3, Notes when Current is Being Injected**, are prohibited. • For self-diagnosis (CNVCLS[2:0] = 3_H) – GCTRL[4:0]: Self-diagnostic Voltage Level 10_H: AVREFH × 1 $0C_H$: AVREFH × 3/4 08_H: AVREFH × 1/2 04_H: AVREFH × 1/4 00_H: AVREFH × 0 Other than above : Setting prohibited Always set the other GCTRL bits to 0. • For normal A/D conversion in addition mode (CNVCLS[2:0] = 4_H) – GCTRL[4:2]: Physical Channel Group These bits specify a physical channel group. – GCTRL[1:0]: Physical sub-channel These bits specify a physical sub-channel. Always set the other GCTRL bits to 0. The count specified by ADDNT is reflected as the number of addition times. • For normal A/D conversion with the MPX (CNVCLS[2:0] = 5_H) – GCTRL[4:0]: MPX Channel Setting These bits set the MPX value to be transferred to an external analog multiplexer. The GCTRL[4:0] value is transferred to ADCBmMPXCURR at the start of virtual channel, and an interrupt request (ADMPXIm) or a DMA request is issued. Activating an interrupt or the DMAC and transferring ADCBmMPXCURR to Pn or PSRn of the I/O port or to CSIHnTX0H of CSIH enables transfer of the MPX value to an external analog multiplexer. For details, see **Section 26.7.5, Example of Operation of External Analog Multiplexer**. • For normal A/D conversion with the MPX in addition mode (CNVCLS[2:0] = 6_H) – GCTRL[4:0]: MPX Channel Setting These bits set the MPX value to be transferred to an external analog multiplexer. The GCTRL[4:0] value is transferred to ADCBmMPXCURR at the start of virtual channel, and an interrupt request (ADMPXIm) or a DMA request is issued. Activating an interrupt or the DMAC and transferring ADCBmMPXCURR to Pn or PSRn of the I/O port or to CSIHnTX0H of CSIH enables transfer of the MPX value to an external analog multiplexer. For details, see **Section 26.7.5, Example of Operation of External Analog Multiplexer**. The count specified by ADDNT is reflected as the number of addition times.

CAUTION

To prevent malfunction, ADCBmVCRn should be set while SGACT for all scan groups is 0 (before scan groups are started), ADSTARTE for all scan groups is 0, TRGMD[0] for scan groups 0, 1, and 2 is 0_H, and TRGMD[1:0] for scan groups 3 and 4 is 0_H.

NOTE

m = 0, 1; n = 0 to 39; x = 0 to 4

26.5.2 ADCBmDRn — Data Register n

ADCBmDRn is a 16-bit read-only register that stores an A/D converted value.

ADCBmDRn is provided for each virtual channel, and its format depends on the DFMT setting of ADCBmADCR2 and the ADDNT setting (when CNVCLS[2:0] of ADCBmVCRn = 4_H or 6_H). The format of data transferred to the DFE (Digital Filter Engine), IFC (Integer/Floating-point Conversion module), and ASF (ADC Summation function) also depends on the above settings. ADCBmDRn is cleared to 0000_H when ADCBmDRn or ADCBmDIRn is read or ADCBmDRn is read via the IFC while RDCLRE is set to 1. ADCBmDRi and ADCBmDRi+1 must always be read together as 32-bit data. When the data must be read in 16-bit units, set RDCLRE to 0. Note that ADCBmDIRi+1.WFLG at that time will be cleared by reading ADCBmDRi in 16-bit units. When CNVCLS[2:0] is 2_H, no data is stored in ADCBmDRn. ADCBmDRn is initialized to 0000_H at a reset.

For signed fixed-point format (DFMT = 0)

	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Convert once	S													0	0	0
Convert twice	S														0	0
Convert 4 times	S															0

↑ Position of decimal point

For signed integer format (DFMT = 1)

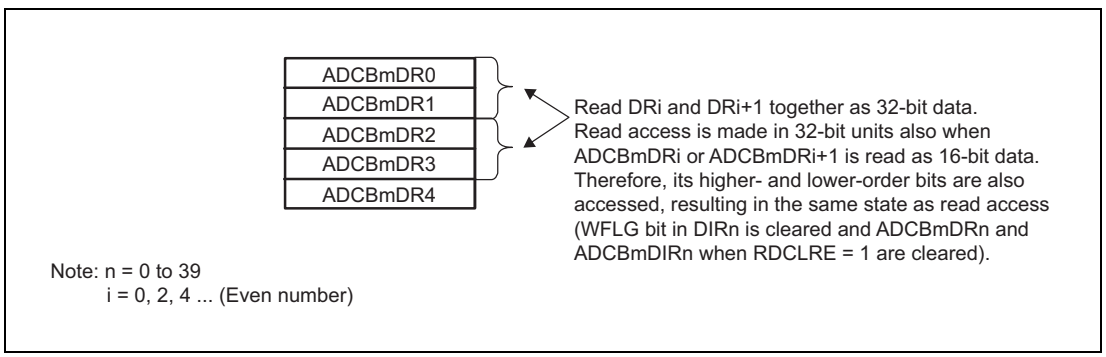
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Convert once	S	S	S	S												
Convert twice	S	S	S													
Convert 4 times	S	S														

↗ Position of decimal point

S	: Sign bit (always 0)
0	: Zero extension

The format setting in ADDNT is valid when CNVCLS[2:0] = 4_H or 6_H.

If CNVCLS[2:0] is not 4_H or 6_H, the format is “convert once.”



NOTE

m = 0, 1; n = 0 to 39; i = 0, 2, 4, ... (Even)

26.5.3 ADCBmDIRn — Data Supplementary Information Register n

ADCBmDIRn is a 32-bit read-only register that stores supplementary information of ADCBmDRn and A/D converted value. ADCBmDIRn is provided for each virtual channel. ADCBmDIRn is cleared to 0000 0000_H when ADCBmDRn or ADCBmDIRn is read or ADCBmDRn is read via the IFC while RDCLRE is set to 1. WFLG is cleared when ADCBmDRn or ADCBmDIRn is read or ADCBmDRn is read via the IFC regardless of the RDCLRE setting. This register must always be read as 32-bit data. ADCBmDRn is read from the lower 16 bits. ADCBmDIRn is initialized to 0000 0000_H at a reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	WFLG	PRTY	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ADCBmDRn															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 26.8 ADCBmDIRn Register Contents

Bit Position	Bit Name	Function
31 to 26	—	Reserved These bits are always read as 0. The write value should always be 0.
25	WFLG	Write Flag Setting condition An A/D converted value is stored in ADCBmDRn. Clearing conditions ADCBmDRn or CBmDIRn is rad. ADCBmDRn is read via IFC. This bit is not set when CNVCLS[2:0] = 2_H.
24	PRTY	Parity Parity bit (even parity) for ADCBmDRn This bit is not set when CNVCLS[2:0] = 2_H.
23 to 16	—	Reserved These bits are always read as 0. The write value should always be 0.

NOTE

m = 0, 1; n = 0 to 39

26.5.4 ADCBmADHALTR — AD Halt Register

ADCBmADHALTR is an 8-bit write-only register that halts the ADC. The register bits are always read as 0.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	HALT
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 26.9 ADCBmADHALTR Register Contents

Bit Position	Bit Name	Function
7 to 1	—	Reserved These bits are always read as 0. The write value should always be 0.
0	HALT	Halt All scan groups and all AD timers are halted and initialized, and the ADC becomes the idle state. Writing 0: Scan groups and timers are not halted. Writing 1: Scan groups and timers are halted.

NOTE

m = 0, 1

26.5.5 ADCBmADCR1 — AD Control Register 1

ADCBmADCR1 is an 8-bit readable/writable register for ADC common control. ADCBmADCR1 is initialized to 00_H at a reset.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	SUSMTD[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 26.10 ADCBmADCR1 Register Contents

Bit Position	Bit Name	Function
7 to 2	—	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	SUSMTD[1:0]	Suspend Method These bits select the suspend method when a higher-priority scan group interrupts a lower-priority scan group. Synchronous suspend: If a request from a higher-priority SG is present while a lower-priority SG is being processed, processing for the lower-priority SG is suspended after the ongoing virtual channel processing is completed, and then processing for the higher-priority SG is executed. After processing for the higher-priority SG is completed, the suspended virtual channel processing for the lower-priority SG is resumed. Asynchronous suspend: If a request from a higher-priority SG is present while a lower-priority SG is being processed, the ongoing virtual channel processing is immediately suspended, and then processing for the higher-priority SG is executed. After processing for the higher-priority SG is completed, the suspended virtual channel processing for the lower-priority SG is resumed. 0_H: Synchronous suspend 1_H: Asynchronous suspend when a higher-priority SG interrupts SG0 Synchronous suspend when a higher-priority SG interrupts a lower-priority SG (except for SG0) 2_H: Asynchronous suspend 3_H: Setting prohibited For details, see **Figure 26.13, Example of Synchronous Suspend and Resume Operation** and **Figure 26.14, Example of Asynchronous Suspend and Resume Operation**.

CAUTION

To prevent malfunction, ADCBmADCR1 should be set while SGACT for all scan groups is 0 (before scan groups are started), ADSTARTE for all scan groups is 0, TRGMD[0] for scan groups 0, 1, and 2 is 0_H, and TRGMD[1:0] for scan groups 3 and 4 is 0_H.

NOTE

m = 0, 1

26.5.6 ADCBmMPXCURCR — MPX Current Control Register

ADCBmMPXCURCR is an 8-bit readable/writable register that controls the ADCBmMPXCURR format. ADCBmMPXCURR is initialized to 00_H at a reset.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	MSKCFMT[3:0]			
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 26.11 ADCBmMPXCURCR Register Contents

Bit Position	Bit Name	Function
7 to 4	—	Reserved These bits are always read as 0. The write value should always be 0.
3 to 0	MSKCFMT[3:0]	MSKC Format Specification These bits specify the MSKC[15:0] format of ADCBmMPXCURR. MSKCFMT[3] 0: MSKC[15:12] = 0000 1: MSKC[15:12] = 1111 MSKCFMT[2] 0: MSKC[11:8] = 0000 1: MSKC[11:8] = 1111 MSKCFMT[1] 0: MSKC[7:4] = 0000 1: MSKC[7:4] = 1111 MSKCFMT[0] 0: MSKC[3:0] = 0000 1: MSKC[3:0] = 1111

CAUTION

To prevent malfunction, ADCBmMPXCURCR should be set while SGACT for all scan groups is 0 (before scan groups are started), ADSTARTE for all scan groups is 0, TRGMD[0] for scan groups 0, 1, and 2 is 0_H, and TRGMD[1:0] for scan groups 3 and 4 is 0_H.

NOTE

m = 0, 1

26.5.7 ADCBmMPXCURR — MPX Current Register

ADCBmMPXCURR is a 32-bit read-only register that stores the MPX value for an external analog multiplexer. ADCBmMPXCURR is initialized to 0000 0000_H at a reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MSKC[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MPXCMD[7:0]								—	—	—	MPXCUR[4:0]				
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 26.12 ADCBmMPXCURR Register Contents

Bit Position	Bit Name	Function
31 to 16	MSKC[15:0]	Mask Control The format depends on the MSKCFMT[3:0] setting of ADCBmMPXCURCR. For details, see **Section 26.5.6, ADCBmMPXCURCR — MPX Current Control Register**.
15 to 8	MPXCMD[7:0]	SPI Communication Command Information A command information register to control an external analog multiplexer by using SPI communication. When a virtual channel in which CNVCLS[2:0] in ADCBmVCRn is set to 5_H or 6_H is started, the MPXCMD[7:0] value in ADCBmMPXCMDR is transferred to MPXCMD[7:0] in ADCBmMPXCURR.
7 to 5	—	Reserved These bits are always read as 0. The write value should always be 0.
4 to 0	MPXCUR[4:0]	Current MPX Value When a virtual channel for which CNVCLS[2:0] in ADCBmVCRn is set to 5_H or 6_H is started, GCTRL[4:0] in ADCBmVCRn is transferred to MPXCUR[4:0]. At this time, an interrupt request to the INTC or a DMA transfer request is generated. The DMAC transfers ADCBmMPXCURR to Pn or PSRn of the I/O port or to CSIHnTX0H of CSIH, enabling the MPX value to be sent to an external analog multiplexer. When Pn is used, transfer the lower 5 bits. When PSRn is used, transfer the MPX value as a 32-bit value. This enables you to rewrite only the necessary ports by using the format control in MSKC[15:0]. When transferring the value to CSIHnTX0H, transfer the lower 16 bits. For details, see **Section 26.7.5, Example of Operation of External Analog Multiplexer**.

NOTE

m = 0, 1; n = 0 to 39

26.5.8 ADCBmMPXOWR — MPX Optional Wait register

ADCBmMPXOWR is an 8-bit readable/writable register that specifies the wait time to be inserted for an external analog multiplexer. ADCBmMPXOWR is initialized to 00_H at a reset.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	MPXOW[3:0]			
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 26.13 ADCBmMPXOWR Register Contents

Bit Position	Bit Name	Function
7 to 4	—	Reserved These bits are always read as 0. The write value should always be 0.
3 to 0	MPXOW[3:0]	MPX Optional Wait These bits specify the wait time to be inserted before A/D conversion is started after a virtual channel for which CNVCLS[2:0] in ADCBmVCRn is 5_H or 6_H is started. 0_H: 0 µs 1_H: 1 µs 2_H: 2 µs 3_H: 3 µs 4_H: 4 µs 5_H: 5 µs 6_H: 6 µs 7_H: 7 µs 8_H: 8 µs 9_H: 9 µs A_H: 10 µs B_H to F_H: Setting prohibited For details, see **Section 26.7.5, Example of Operation of External Analog Multiplexer**.

CAUTION

To prevent malfunction, ADCBmMPXOWR should be set while SGACT for all scan groups is 0 (before scan groups are started), ADSTARTE for all scan groups is 0, TRGMD[0] for scan groups 0, 1, and 2 is 0_H, and TRGMD[1:0] for scan groups 3 and 4 is 0_H.

NOTE

m = 0, 1; n = 0 to 39

26.5.9 ADCBmMPXCMDR — MPX Command Information Register

ADCBmMPXCMDR is an 8-bit readable/writable register that stores SPI communication command information to be transferred to an external analog multiplexer. ADCBmMPXCMDR is initialized to 00_H at a reset.

Bit	7	6	5	4	3	2	1	0
	MPXCMD[7:0]							
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 26.14 ADCBmMPXCMDR Register Contents

Bit Position	Bit Name	Function
7 to 0	MPXCMD[7:0]	SPI Communication Command Information These bits are a register that stores command information for controlling an external analog multiplexer by using the SPI communication. When a virtual channel for which CNVCLS[2:0] in ADCBmVCRn is set to 5_H or 6_H is started, MPXCMD[7:0] in ADCBmMPXCMDR is transferred to MPXCMD[7:0] in ADCBmMPXCURR and it can be read by ADCBmMPXCURR together with MPXCUR[4:0].

NOTE

m = 0, 1 : n = 0 to 39

26.5.10 ADCBmADCR2 — AD Control Register 2

ADCBmADCR2 is an 8-bit readable/writable register for ADC common control. ADCBmADCR2 is initialized to 00_H at a reset.

Bit	7	6	5	4	3	2	1	0
	—	—	—	DFMT	—	—	—	ADDNT
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R	R	R	R/W

Table 26.15 ADCBmADCR2 Register Contents

Bit Position	Bit Name	Function
7 to 5	—	Reserved These bits are always read as 0. The write value should always be 0.
4	DFMT	Data Format 0: Signed fixed-point format 1: Signed integer format This bit specifies the format of data to be transferred to ADCBmDRn, DFE, and ASF. For details of data format, see **Section 26.5.2, ADCBmDRn — Data Register n**.
3 to 1	—	Reserved These bits are always read as 0. The write value should always be 0.
0	ADDNT	Addition Count Select 0: Add twice 1: Add 4 times This register is valid only when CNVCLS[2:0] is 4_H or 6_H.

CAUTION

To prevent malfunction, perform ADCBmADCR2 settings when SGACT of all scan groups is 0 (before scan groups are started), ADSTARTE of all scan groups is 0, TRGMD[0] of scan groups 0, 1, and 2 is 0_H, and TRGMD[1:0] of scan groups 3 and 4 is 0_H.

NOTE

m = 0, 1 : n = 0 to 39

26.5.11 ADCBmDFASENTSGER — DFE/ASF Entry Scan Group Enable Register

ADCBmDFASENTSGER is a 16-bit readable/writable register that enables or disables scan groups to be entered to the DFE and ASF. ADCBmDFASENTSGER is initialized to 0000_H at a reset.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	DFENTSG4E	DFENTSG3E	DFENTSG2E	DFENTSG1E	DFENTSG0E	—	—	—	ASENTSG4E	ASENTSG3E	ASENTSG2E	ASENTSG1E	ASENTSG0E
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Table 26.16 ADCBmDFASENTSGER Register Contents

Bit Position	Bit Name	Function
15 to 13	—	Reserved These bits are always read as 0. The write value should always be 0.
12 to 8	DFENTSGxE	DFE Entry Scan Group Enable 0: Entry to DFE is disabled when starting SGx. 1: Entry to DFE is enabled when starting SGx. Entry is made in virtual channels for which DFENT in ADCBmVCRn is set to 1.
7 to 5	—	Reserved These bits are always read as 0. The write value should always be 0.
4 to 0	ASENTSGxE	ASF Entry Scan Group Enable 0: Entry to ASF is disabled when starting SGx. 1: Entry to ASF is enabled when starting SGx. Entry is made in virtual channels for which DFENT in ADCBmVCRn is set to 1. Use the ASENTSGxE bits by setting one of these bits to 1.

CAUTIONS

1. To prevent malfunction, ADCBmDFASENTSGER should be set while SGACT for all scan groups is 0 (before scan groups are started), ADSTARTE for all scan groups is 0, TRGMD[0] for scan groups 0, 1, and 2 is 0_H, and TRGMD[1:0] for scan groups 3 and 4 is 0_H.
2. Do not set DFENTSGxE or ASENTSGxE for the same scan group to 1. If it is enabled for the same scan group, entry is made to both DFE and ASF of virtual channels for which DFENT in ADCBmVCRn is set to 1 specified for the scan group.

NOTE

m = 0, 1; n = 0 to 39; x = 0 to 4

26.5.12 ADCBmADENDP — A/D Conversion Monitor Virtual Channel Pointer

ADCBmADENDP is an 8-bit readable/writable register that selects a virtual channel that outputs the A/D conversion timing to ADENDm. ADCBmADENDP is initialized to 00_H at a reset.

Bit	7	6	5	4	3	2	1	0
	—	—	ENDP[5:0]					
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Table 26.17 ADCBmADENDP Register Contents

Bit Position	Bit Name	Function
7, 6	—	Reserved These bits are always read as 0. The write value should always be 0.
5 to 0	ENDP[5:0]	A/D Conversion Monitor Virtual Channel Pointer When the virtual channel selected by ADCBmADENDP is started, a high level is output to ADENDm. When the virtual channel selected by ADCBmADENDP ends, a low level is output.

CAUTION

To prevent malfunction, ADCBmADENDP should be set while SGACT for all scan groups is 0 (before scan groups are started), ADSTARTE for all scan groups is 0, TRGMD[0] for scan groups 0, 1, and 2 is 0_H, and TRGMD[1:0] for scan groups 3 and 4 is 0_H.

NOTE

m = 0, 1

26.5.13 ADCBmSFTCR — Safety Control Register

ADCBmSFTCR is an 8-bit readable/writable register for safety control. ADCBmSFTCR is initialized to 00_H at a reset.

Bit	7	6	5	4	3	2	1	0
	—	—	—	RDCLRE	ULEIE	OWEIE	PEIE	IDEIE
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Table 26.18 ADCBmSFTCR Register Contents

Bit Position	Bit Name	Function
7 to 5	—	Reserved These bits are always read as 0. The write value should always be 0.
4	RDCLRE	Read and Clear Enable 0: ADCBmDRn and ADCBmDIRn are not cleared by reading ADCBmDRn or ADCBmDIRn and by reading ADCBmDRn via the IFC. 1: ADCBmDRn and ADCBmDIRn are cleared by reading ADCBmDRn or ADCBmDIRn and by reading ADCBmDRn via the IFC. **NOTE** WFLG in ADCBmDIRn is cleared by reading ADCBmDRn or ADCBmDIRn and by reading ADCBmDRn via the IFC regardless of the RDCLRE setting.
3	ULEIE	Upper-Limit/Lower-Limit Error Interrupt Enable 0: Disabled 1: Enabled
2	OWEIE	Overwrite Error Interrupt Enable 0: Disabled 1: Enabled
1	PEIE	Parity Error Interrupt Enable 0: Disabled 1: Enabled
0	IDEIE	ID Error Interrupt Enable 0: Disabled 1: Enabled

CAUTION

To prevent malfunction, ADCBmSFTCR should be set while SGACT for all scan groups is 0 (before scan groups are started), ADSTARTE for all scan groups is 0, TRGMD[0] for scan groups 0, 1, and 2 is 0_H, and TRGMD[1:0] for scan groups 3 and 4 is 0_H.

NOTE

m = 0, 1; n = 0 to 39

26.5.14 ADCBmTDCR — Pin-Level Self-Diagnosis Control Register

ADCBmTDCR is an 8-bit readable/writable register that controls the pin-level self-diagnosis. ADCBmTDCR is initialized to 00_H at a reset.

Bit	7	6	5	4	3	2	1	0
	TDE	—	—	—	—	—	TDLV[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R	R	R	R/W	R/W

Table 26.19 ADCBmTDCR Register Contents

Bit Position	Bit Name	Function
7	TDE	Pin-Level Self-Diagnosis 0: Pin-level self-diagnosis is disabled. 1: Pin-level self-diagnosis is enabled. When TDE is set to 1, all analog pins are disconnected from the input buffer. When TDE is set to 0, all analog pins are connected to the input buffer. When TDE is set to 1, the voltage is fixed to the level specified by TDLV[1:0]. Performing A/D conversion in this state and checking the A/D converted value allows diagnosis of the path from an analog pin to the ADC.
6 to 2	—	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	TDLV[1:0]	Pin-Level Self-Diagnosis 0_H: Even numbers of physical channel groups are discharged to AVSS, and odd numbers of physical channel groups are charged to AVSS. 1_H: Even numbers of physical channel groups are charged to AVCC, and odd numbers of physical channel groups are discharged to AVSS. 2_H: Even numbers of physical channel groups are discharged to AVSS, and odd numbers of physical channel groups are charged to 1/2*AVCC. 3_H: Even numbers of physical channel groups are charged to 1/2*AVCC, and odd numbers of physical channel groups are discharged to AVSS.

CAUTION

To prevent malfunction, ADCBmTDCR should be set while SGACT for all scan groups is 0 (before scan groups are started), ADSTARTE for all scan groups is 0, TRGMD[0] for scan groups 0, 1, and 2 is 0_H, and TRGMD[1:0] for scan groups 3 and 4 is 0_H.

NOTE

m = 0, 1

26.5.15 ADCBmODCR — Wiring-break Detection Control Register

ADCBmODCR is an 8-bit readable/writable register that controls wiring-break detection.

ADCBmODCR is initialized to 00_H at a reset.

Bit	7	6	5	4	3	2	1	0
	ODE	—	ODPW[5:0]					
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R	R/W	R/W	R/W	R/W	R/W	R/W

Table 26.20 ADCBmODCR Register Contents

Bit Position	Bit Name	Function
7	ODE	Wiring-break Detection Enable 0: Wiring-break is not detected. 1: Wiring-break is detected. When ODE is set to 1, wiring-break detection is enabled for all analog pins. After sampling for A/D conversion ends, analog pins for which A/D conversion is to be performed are discharged with the pulse width specified in ODPW[5:0].
6	—	Reserved This bit is always read as 0. The write value should always be 0.
5 to 0	ODPW[5:0]*1	Wiring-break Detection Pulse Width 04_H: 1 state (of internal clock) 05_H: 2 states (of internal clock) : 13_H: 16 states (of internal clock) 14_H: 17 states (of internal clock)

Note 1. When ODE = 1 (wiring-break detection is enabled), the setting of ODPW[5:0] must be greater than 03_H and less than 15_H.

CAUTIONS

1. To prevent malfunction, ADCBmODCR should be set while SGACT for all scan groups is 0 (before scan groups are started), ADSTARTE for all scan groups is 0, TRGMD[0] for scan groups 0, 1, and 2 is 0_H, and TRGMD[1:0] for scan groups 3 and 4 is 0_H.
2. Do not perform T&H (CNVCLS in ADCBmVCRn is 2_H) or A/D conversion of the hold value (CNVCLS in ADCBmVCRn is 1_H) with wiring-break detection enabled (ODE = 1).

NOTE

m = 0, 1; n = 0 to 39

26.5.16 ADCBmULLMTBR0 to 2 — Upper Limit/Lower Limit Table Register 0 to 2

ADCBmULLMTBR0 to ADCBmULLMTBR2 are 16/32-bit readable/writable registers that set the upper-limit and lower-limit values of an A/D converted value. Specify any of ADCBmULLMTBR0 to ADCBmULLMTBR2 by ULS[1:0] in ADCBmULLMSRx. These registers are initialized to 7FFE 0000_H by a reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ULMTB[15:0]															
Value after reset	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LLMTB[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R

Table 26.21 ADCBmULLMTBR Register Contents

Bit Position	Bit Name	Function
31 to 16	ULMTB[15:0]	Upper Limit Table These bits specify the upper-limit value of an A/D converted value. ULE (upper limit/lower limit error) is set when the following condition is met. ULMTB[15:0] < A/D converted value The ULMTB[15:0] format is the signed fixed-point format regardless of the format of ADCBmDRn. If the signed integer format is selected for the ADCBmDRn format, the ADCBmDRn format is replaced with the signed fixed-point format, and then the values are compared. Note that ULMTB[15] and ULMTB[0] are always fixed to 0.
15 to 0	LLMTB[15:0]	Lower Limit Table These bits specify the lower-limit value of an A/D converted value. ULE (upper limit/lower limit error) is set when the following condition is met. LLMTB[15:0] > A/D converted value The LLMTB[15:0] format is the signed fixed-point format regardless of the format of ADCBmDRn. If the signed integer format is selected for the ADCBmDRn format, the ADCBmDRn format is replaced with the signed fixed-point format, and then the values are compared. Note that LLMTB[15] and LLMTB[0] are always fixed to 0.

CAUTION

To prevent malfunction, ADCBmULLMTBR0 to ADCBmULLMTBR2 should be set while SGACT for all scan groups is 0 (before scan groups are started), ADSTARTE for all scan groups is 0, TRGMD[0] for scan groups 0, 1, and 2 is 0_H, and TRGMD[1:0] for scan groups 3 and 4 is 0_H.

NOTE

m = 0, 1; n = 0 to 39; x = 0 to 4

26.5.17 ADCBmECR — Error Clear Register

ADCBmECR is an 8-bit write-only register that controls error clear. The register bits are always read as 0.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	ULEC	OWEC	PEC	IDEC
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	W	W	W	W

Table 26.22 ADCBmECR Register Contents

Bit Position	Bit Name	Function
7 to 4	—	Reserved These bits are always read as 0. The write value should always be 0.
3	ULEC	Upper Limit/Lower Limit Error Clear Writing 0: Not cleared Writing 1: Cleared
2	OWEC	Overwrite Error Clear Writing 0: Not cleared Writing 1: Cleared
1	PEC	Parity Error Clear Writing 0: Not cleared Writing 1: Cleared
0	IDEC	ID Error Clear Writing 0: Not cleared Writing 1: Cleared

NOTE

m = 0, 1

26.5.18 ADCBmULER — Upper Limit/Lower Limit Error Register

ADCBmULER is an 8-bit read-only register that indicates upper limit/lower limit errors.

ADCBmULER is initialized to 00_H at a reset.

Bit	7	6	5	4	3	2	1	0
	ULE	—	ULECAP[5:0]					
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 26.23 ADCBmULER Register Contents

Bit Position	Bit Name	Function
7	ULE	Upper Limit/Lower Limit Error 0: No error 1: An error is present. Setting condition The A/D converted value exceeds the range of the specified upper limit/lower limit table. Clearing condition A value of 1 is written to ULEC.
6	—	Reserved This bit is always read as 0. The write value should always be 0.
5 to 0	ULECAP[5:0]	Upper Limit/Lower Limit Error Capture The virtual channel at the time when an upper limit/lower limit error occurred is captured. Capturing condition ULE = 0 and the A/D converted value exceeds the range of the specified upper limit/lower limit table. Clearing condition A value of 1 is written to ULEC.

CAUTION

ADCBmULER is updated when the A/D converted value is written to ADCBmDRn.

NOTE

m = 0, 1; n = 0 to 39

26.5.19 ADCBmOWER — Overwrite Error Register

ADCBmOWER is an 8-bit read-only register that indicates an overwrite error.

ADCBmOWER is initialized to 00_H at a reset.

Bit	7	6	5	4	3	2	1	0
	OWE	—	OWECAP[5:0]					
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 26.24 ADCBmOWER Register Contents

Bit Position	Bit Name	Function
7	OWE	Overwrite Error 0: No error 1: An error is present. Setting condition The A/D converted value is written to ADCBmDRn when WFLG = 1. Clearing condition A value of 1 is written to OWEC.
6	—	Reserved This bit is always read as 0. The write value should always be 0.
5 to 0	OWECAP[5:0]	Overwrite Error Capture The virtual channel at the time when an overwrite error occurred is captured. Capturing condition The A/D converted value is written to ADCBmDRn when OWE = 0 and WFLG = 1. Clearing condition A value of 1 is written to OWEC.

CAUTION

ADCBmOWER is updated when the A/D converted value is written to ADCBmDRn.

NOTE

m = 0, 1; n = 0 to 39

26.5.20 ADCBmPER — Parity Error Register

ADCBmPER is an 8-bit read-only register that indicates a parity error. ADCBmPER is initialized to 00_H at a reset.

Bit	7	6	5	4	3	2	1	0
	PE	—	PECAP[5:0]					
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 26.25 ADCBmPER Register Contents

Bit Position	Bit Name	Function
7	PE	Parity Error 0: No error 1: An error is present. Setting condition A parity error is detected. Clearing condition A value of 1 is written to PEC.
6	—	Reserved This bit is always read as 0. The write value should always be 0.
5 to 0	PECAP[5:0]	Parity Error Capture The virtual channel at the time when a parity error occurred is captured. Capturing condition A parity error is detected when PE = 0. Clearing condition A value of 1 is written to PEC.

CAUTION

ADCBmPER is updated when ADCBmDRn or ADCBmDIRn is read, but is not updated when ADCBmDRn is read via the IFC.

NOTE

m = 0, 1; n = 0 to 39

26.5.21 ADCBmIDER — ID Error Register

ADCBmIDER is an 8-bit read-only register that indicates an ID error. ADCBmIDER is initialized to 00_H at a reset.

Bit	7	6	5	4	3	2	1	0
	IDE	—	IDECAP[5:0]					
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 26.26 ADCBmIDER Register Contents

Bit Position	Bit Name	Function
7	IDE	ID Error 0: No error 1: An error is present. Setting condition The physical channel (ADC0: AN043 and ADC1: AN100 when external multiplexer is specified) specified in ADCBmVCRn does not match the physical channel actually converted. Clearing condition A value of 1 is written to IDEC.
6	—	Reserved This bit is always read as 0. The write value should always be 0.
5 to 0	IDECAP[5:0]	ID Error Capture The virtual channel at the time when an ID error occurred is captured. Capturing condition The physical channel (ADC0: AN043 and ADC1: AN100 when external multiplexer is specified) specified in ADCBmVCRn does not match the physical channel actually converted when IDE = 0. Clearing condition A value of 1 is written to IDEC.

CAUTION

ADCBmIDER is updated when the A/D converted value is written to ADCBmDRn.

NOTE

m = 0, 1; n = 0 to 39

26.6 Scan Group Specific Registers

This section describes registers provided for each scan group.

26.6.1 ADCBmSGSTCRx — Scan Group x Start Control Register

ADCBmSGSTCRx is an 8-bit write-only register that controls the start of scan group x. The register bits are always read as 0.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	SGST
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 26.27 ADCBmSGSTCRx Register Contents

Bit Position	Bit Name	Function
7 to 1	—	Reserved These bits are always read as 0. The write value should always be 0.
0	SGST	Scan Group Start Condition for starting scan group x: A value of 1 is written to SGST when SGACT = 0

NOTE

m = 0,1; x = 0 to 4

26.6.2 ADCBmADTSTCRy — A/D Timer y Start Control Register

ADCBmADTSTCRy is an 8-bit write-only register that controls the start of A/D timer y. The register bits are always read as 0.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ADTST
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 26.28 ADCBmADTSTCRy Register Contents

Bit Position	Bit Name	Function
7 to 1	—	Reserved These bits are always read as 0. The write value should always be 0.
0	ADTST	A/D Timer Start Condition for starting A/D timer y: A value of 1 is written to ADTST when ADTACT = 0

NOTE

m = 0, 1; y = 3, 4

26.6.3 ADCBmADTENDCRy — A/D Timer y End Control Register

ADCBmADTENDCRy is an 8-bit write-only register that controls the end of A/D timer y. The register bits are always read as 0.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ADTEND
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 26.29 ADCBmADTENDCRy Register Contents

Bit Position	Bit Name	Function
7 to 1	—	Reserved These bits are always read as 0. The write value should always be 0.
0	ADTEND	A/D Timer End Condition for finishing A/D timer y: A value of 1 is written to ADTEND when ADTACT = 1

NOTE

m = 0, 1; y = 3, 4

26.6.4 ADCBmSGCRx — Scan Group x Control Register

ADCBmSGCRx is an 8-bit readable/writable register that controls scan group x. ADCBmSGCRx is initialized to 00_H at a reset.

- When x = 0 to 2

Bit	7	6	5	4	3	2	1	0
	—	ADSTARTE	SCANMD	ADIE	—	—	—	TRGMD[0]
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R	R	R	R/W

Table 26.30 ADCBmSGCRx Register Contents (x = 0 to 2)

Bit Position	Bit Name	Function
7	—	Reserved This bit is always read as 0. The write value should always be 0.
6	ADSTARTE	Scan Group Synchronization Start Enable 0: ADSTART is disabled. 1: ADSTART is enabled.
5	SCANMD	Scan Mode 0: Multicycle scan mode 1: Continuous scan mode In multicycle scan mode, scans are repeated as many times as specified in ADCBmSGMCYCRx. In continuous scan mode, scans are repeated with no limit of times.
4	ADIE	Scan End Interrupt Enable 0: ADImx is not output at the end of scan for SGx. 1: ADImx is output at the end of scan for SGx. ADIE of ADCBmSGCRx is independent of ADIE of ADCBmVCRn. For details, see **Section 26.7.16, Scan End Interrupt Request**.
3 to 1	—	Reserved These bits are always read as 0. The write value should always be 0.
0	TRGMD[0]	Trigger Mode 0_H: Trigger input to SGx is disabled. 1_H: The SGx_TRG hardware trigger is selected for the trigger input to SGx. TRGMD is for enabling or disabling of hardware trigger. Software trigger is always enabled.

- When x = 3 or 4

Bit	7	6	5	4	3	2	1	0
	ADTSTARTE	ADSTARTE	SCANMD	ADIE	—	—	TRGMD[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W

Table 26.31 ADCBmSGCRx Register Contents (x = 3 or 4)

Bit Position	Bit Name	Function
7	ADTSTARTE	A/D Timer Synchronization Start Enable 0: ADSTART is disabled. 1: ADSTART is enabled.
6	ADSTARTE	Scan Group Synchronization Start Enable 0: ADSTART is disabled. 1: ADSTART is enabled.
5	SCANMD	Scan Mode 0: Multicycle scan mode 1: Continuous scan mode In multicycle scan mode, scans are repeated as many times as specified in ADCBmSGMCYCRx. In continuous scan mode, scans are repeated with no limit of times.
4	ADIE	Scan End Interrupt Enable 0: ADImx is not output at the end of scan for SGx. 1: ADImx is output at the end of scan for SGx. ADIE of ADCBmSGCRx is independent of ADIE of ADCBmVCRn. For details, see **Section 26.7.16, Scan End Interrupt Request**.
3, 2	—	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	TRGMD[1:0]	Trigger Mode 0_H: Trigger input to SGx is disabled. Trigger input to A/D timer x is disabled. 1_H: SGx_TRG hardware trigger is selected for the trigger input to SGx. Trigger input to A/D timer x is disabled. 2_H: A/D timer trigger x is selected for the trigger input to SGx. Trigger input to A/D timer x is disabled. 3_H: A/D timer trigger x is selected for the trigger input to SGx. The SGx_TRG hardware trigger is selected for the trigger input to A/D timer x. TRGMD is for enabling or disabling of hardware trigger. Software trigger is always enabled.

CAUTIONS

1. To prevent malfunction, SCANMD and ADIE in ADCBmSGCRx should be set while SGACT for scan group x is 0 (before scan groups are started), ADSTARTE for scan group x is 0, and TRGMD for scan group x is 0_H.
2. If a trigger of a lower-priority scan group is input to a scan group for which continuous scan mode is set (SCANMD = 1_H), the trigger is not accepted. Therefore, it is assumed that continuous scan mode is set for scan group 0.

NOTE

m = 0, 1; n = 0 to 39; x = 0 to 4

26.6.5 ADCBmSGVCSPx — Scan Group x Start Virtual Channel Pointer

ADCBmSGVCSPx is an 8-bit readable/writable register that specifies the start pointer of a virtual channel. ADCBmSGVCSPx is initialized to 00_H at a reset.

Bit	7	6	5	4	3	2	1	0
	—	—	VCSP[5:0]					
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Table 26.32 ADCBmSGVCSPx Register Contents

Bit Position	Bit Name	Function
7, 6	—	Reserved These bits are always read as 0. The write value should always be 0.
5 to 0	VCSP[5:0]*1, *2	Start Virtual Channel Pointer These bits select the virtual channel from which the scan is to be started. When SGx is started, processing for the virtual channels from ADCBmSGVCSPx to ADCBmSGVCEPx is executed.

CAUTIONS

1. ADCBmSGVCSPx must be equal to or less than ADCBmSGVCEPx.
2. To prevent malfunction, ADCBmSGVCSPx should be set while SGACT for scan group x is 0 (before scan groups are started), ADSTARTE for scan group x is 0, and TRGMD for scan group x is 0_H.

NOTE

m = 0, 1; x = 0 to 4

26.6.6 ADCBmSGVCEPx — Scan Group x End Virtual Channel Pointer

ADCBmSGVCEPx is an 8-bit readable/writable register that specifies the end pointer of a virtual channel. ADCBmSGVCEPx is initialized to 00_H at a reset.

Bit	7	6	5	4	3	2	1	0
	—	—	VCEP[5:0]					
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Table 26.33 ADCBmSGVCEPx Register Contents

Bit Position	Bit Name	Function
7, 6	—	Reserved These bits are always read as 0. The write value should always be 0.
5 to 0	VCEP[5:0]*1, *2	End Virtual Channel Pointer These bits select the virtual channel at which the scan is to be ended. When SGx is started, processing for the virtual channels from ADCBmSGVCSPx to ADCBmSGVCEPx is executed.

CAUTIONS

1. ADCBmSGVCSPx must be equal to or smaller than ADCBmSGVCEPx.
2. To prevent malfunction, ADCBmSGVCEPx should be set while SGACT for scan group x is 0 (before scan groups are started), ADSTARTE for scan group x is 0, and TRGMD for scan group x is 0_H.

NOTE

m = 0, 1; x = 0 to 4

26.6.7 ADCBmSGMCYCRx — Scan Group x Multicycle Register

ADCBmSGMCYCRx is an 8-bit readable/writable register that specifies the number of scan times in multicycle scan mode. ADCBmSGMCYCRx is initialized to 00_H at a reset.

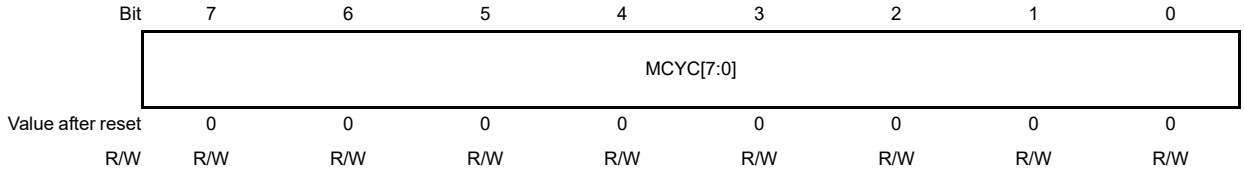


Table 26.34 ADCBmSGMCYCRx Register Contents

Bit Position	Bit Name	Function
7 to 0	MCYC[7:0]	Multicycle Specification These bits specify the number of scan times in multicycle scan mode. Number of scan times = MCYC[7:0] + 1 When SGx is started, scans are repeated for virtual channels from ADCBmSGVCSPx to ADCBmSGVCEPx as many times as specified in ADCBmSGMCYCRx.

CAUTION

To prevent malfunction, ADCBmSGMCYCRx should be set while SGACT for scan group x is 0 (before scan groups are started), ADSTARTE for scan group x is 0, and TRGMD for scan group x is 0_H.

NOTE

m = 0,1; x = 0 to 4

26.6.8 ADCBmSGSRx — Scan Group x Status Register

ADCBmSGSRx is an 8-bit read-only register that indicates the status of scan group x. ADCBmSGSRx is initialized to 00_H at a reset.

- When x = 0 to 2

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	SGACT	—
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 26.35 ADCBmSGSRx Register Contents (x = 0 to 2)

Bit Position	Bit Name	Function
7 to 2	—	Reserved These bits are always read as 0. The write value should always be 0.
1	SGACT	Scan Group Status 0: There is no source in SGx. 1: There is a source in SGx.
0	—	Reserved This bit is always read as 0. The write value should always be 0.

- When x = 3 or 4

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	ADTACT	SGACT	—
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 26.36 ADCBmSGSRx Register Contents (x = 3, 4)

Bit Position	Bit Name	Function
7 to 3	—	Reserved These bits are always read as 0. The write value should always be 0.
2	ADTACT	A/D Timer Status 0: A/D timer x is in idle state. 1: A/D timer x is running.
1	SGACT	Scan Group Status 0: There is no source in SGx. 1: There is a source in SGx.
0	—	Reserved This bit is always read as 0. The write value should always be 0.

NOTE

m= 0, 1; x = 0 to 4

26.6.9 ADCBmADTIPRy — A/D Timer Initial Phase Register y

ADCBmADTIPRy is a 32-bit readable/writable register that sets the initial phase of A/D timer y. ADCBmADTIPRy is initialized to 0000 0000_H at a reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	ADTIP[20:16]				
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ADTIP[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 26.37 ADCBmADTIPRy Register Contents

Bit Position	Bit Name	Function
31 to 21	—	Reserved These bits are always read as 0. The write value should always be 0.
20 to 0	ADTIP[20:0]	A/D Timer Initial Phase These bits set the initial phase of A/D timer y. (1) After A/D timer y is started, ADCBmADTIPRy is loaded to A/D timer y and the timer counts down. (2) After A/D timer y becomes 0, A/D timer trigger y is output for one cycle, ADCBmADTPRRy is loaded to A/D timer y, and the timer counts down again. After that, (2) is repeated. For details, see **Section 26.7.8, Example of A/D Timer Operation**.

CAUTION

To prevent malfunction, ADCBmADTIPRy should be set while ADTACT for scan group y is 0 (before A/D timer is started), ADTSTARTE for scan group y is 0, and TRGMD[1:0] for scan group y is not 3_H.

NOTE

m = 0, 1; y = 3, 4

26.6.10 ADCBmADTPRRy — A/D Timer Cycle Register y

ADCBmADTPRRy is a 32-bit readable/writable register that sets the cycle of A/D timer y.
ADCBmADTPRRy is initialized to 001F FFFF$_H$ at a reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	ADTPR[20:16]				
Value after reset	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ADTPR[15:0]															
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 26.38 ADCBmADTPRRy Register Contents

Bit Position	Bit Name	Function
31 to 21	—	Reserved These bits are always read as 0. The write value should always be 0.
20 to 0	ADTPR[20:0]	A/D Timer Cycle These bits set the cycle of A/D timer y. (1) After A/D timer y is started, ADCBmADTIPRy is loaded to A/D timer y and the timer counts down. (2) After A/D timer y becomes 0, A/D timer trigger y is output for one cycle, ADCBmADTPRRy is loaded to A/D timer y, and the timer counts down again. After that, (2) is repeated. For details, see **Section 26.7.8, Example of A/D Timer Operation**.

CAUTION

To prevent malfunction, ADTPPRy should be set while ADTACT for scan group y is 0 (before A/D timer is started), ADTSTARTE for scan group y is 0, and TRGMD[1:0] for scan group y is not 3$_H$.

NOTE

m = 0, 1; y = 3, 4

26.6.11 ADCBmULLMSRx — Scan Group x Upper Limit/Lower Limit Table Select Register

ADCBmULLMSRx is an 8-bit readable/writable register that controls scan group x. ADCBmSGCRx is initialized to 00_H at a reset.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	ULS[1:0]	
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 26.39 ADCBmULLMSRx Register Contents

Bit Position	Bit Name	Function
7 to 2	—	Reserved These bits are always read as 0. The write value should always be 0.
1, 0	ULS[1:0]	Upper Limit/Lower Limit Table Select 0_H: Neither upper limit nor lower limit is checked. 1_H: Upper limit and lower limit are checked in ADCBmULLMTBR0. 2_H: Upper limit and lower limit are checked in ADCBmULLMTBR1. 3_H: Upper limit and lower limit are checked in ADCBmULLMTBR2. Upper limit and lower limit are checked by using the upper limit/lower limit table selected by ULS[1:0] when storing the A/D converted value in ADCBmDRn.

CAUTION

To prevent malfunction, ADCBmULLMSRx should be set while SGACT for scan group x is 0 (before scan groups are started), ADSTARTE for scan group x is 0, and TRGMD for scan group x is 0_H.

NOTE

m = 0, 1: n = 0 to 39; x = 0 to 4

26.7 Operation

26.7.1 Setting Procedure

26.7.1.1 Initial Settings

A/D conversion of the ADC starts by setting the registers shown in **Figure 26.3**. For trigger input, see **Section 26.7.1.2, Trigger Input Flow**.

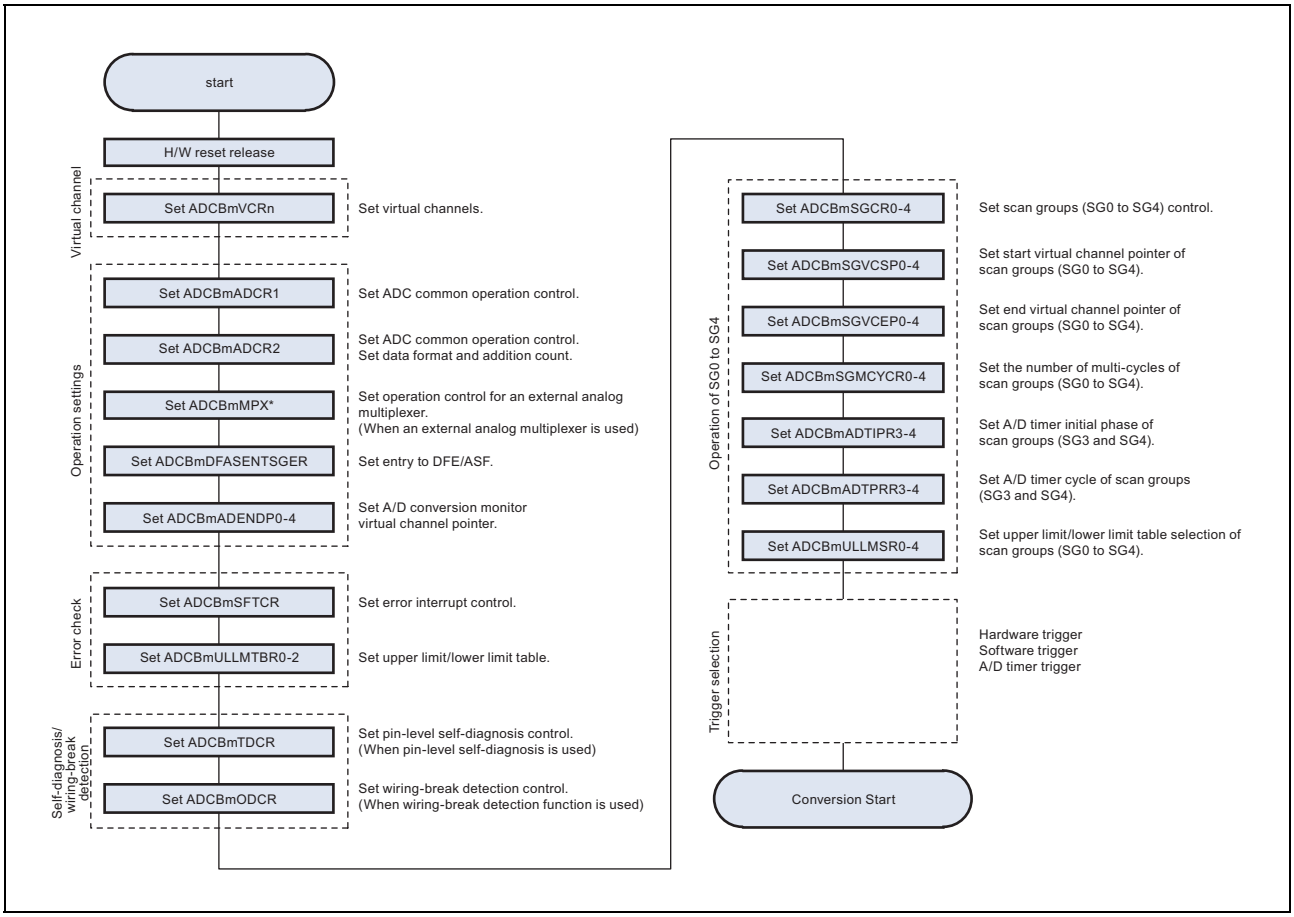


Figure 26.3 Initial Settings

NOTE

m = 0, 1 : n = 0 to 39

26.7.1.2 Trigger Input Flow

A/D conversion start triggers of the ADC include hardware triggers, software triggers, and A/D timer triggers. Scan group x (x = 0 to 4) supports the software trigger specified by ADSTART and SGST (SG0 to SG4), the hardware trigger of SGx_TRG (x = 0 to 4), and the A/D timer trigger (SG3 and SG4).

A/D conversion starts according to the trigger input flows shown in **Figure 26.4** and **Figure 26.5**. For the trigger configuration of SGx_TRG (x = 0 to 4), see **26.7.11, Selection of Trigger Input for a Scan Group**. For the initial settings flow shown in these figures, see **Section 26.7.1.1, Initial Settings**.

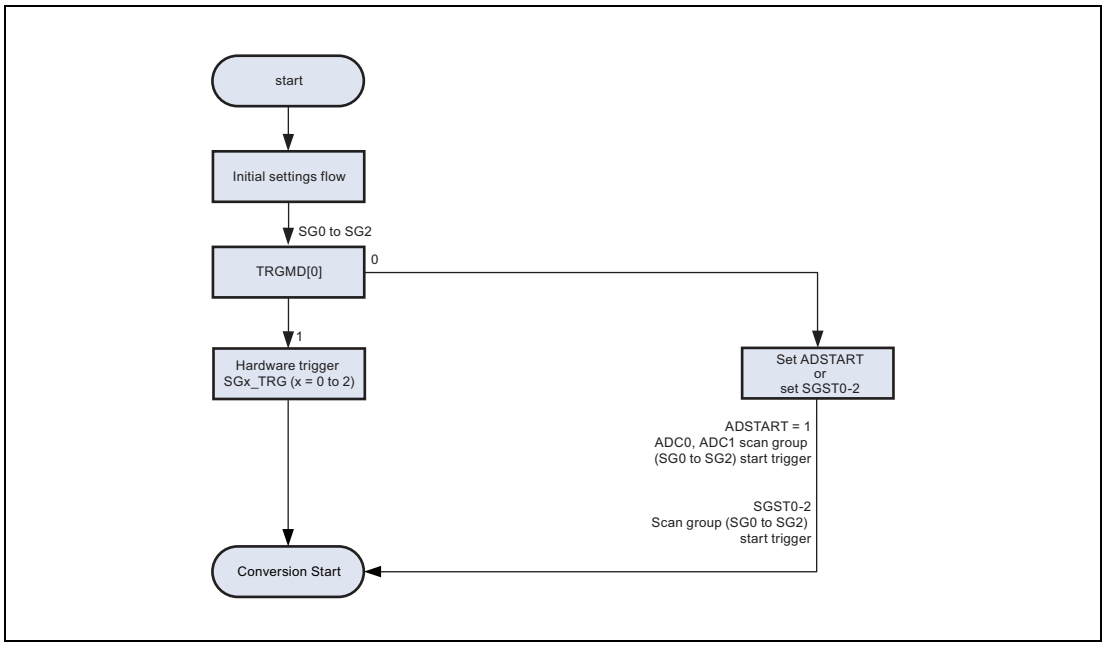


Figure 26.4 Trigger Input Flow (SG0 to SG2)

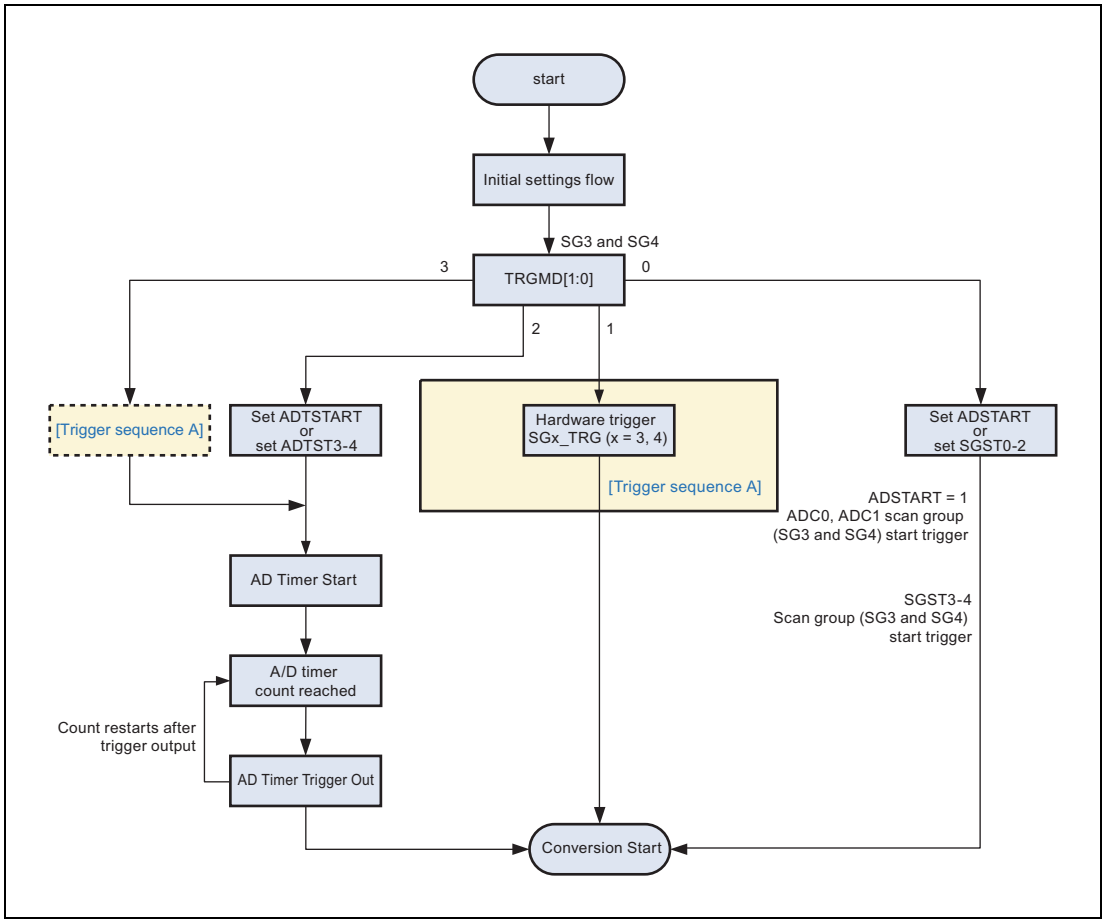


Figure 26.5 Trigger Input Flow (SG3 and SG4)

26.7.1.3 Terminating Procedure

The ADC is forcibly terminated according to the processing flow shown in **Figure 26.6**.

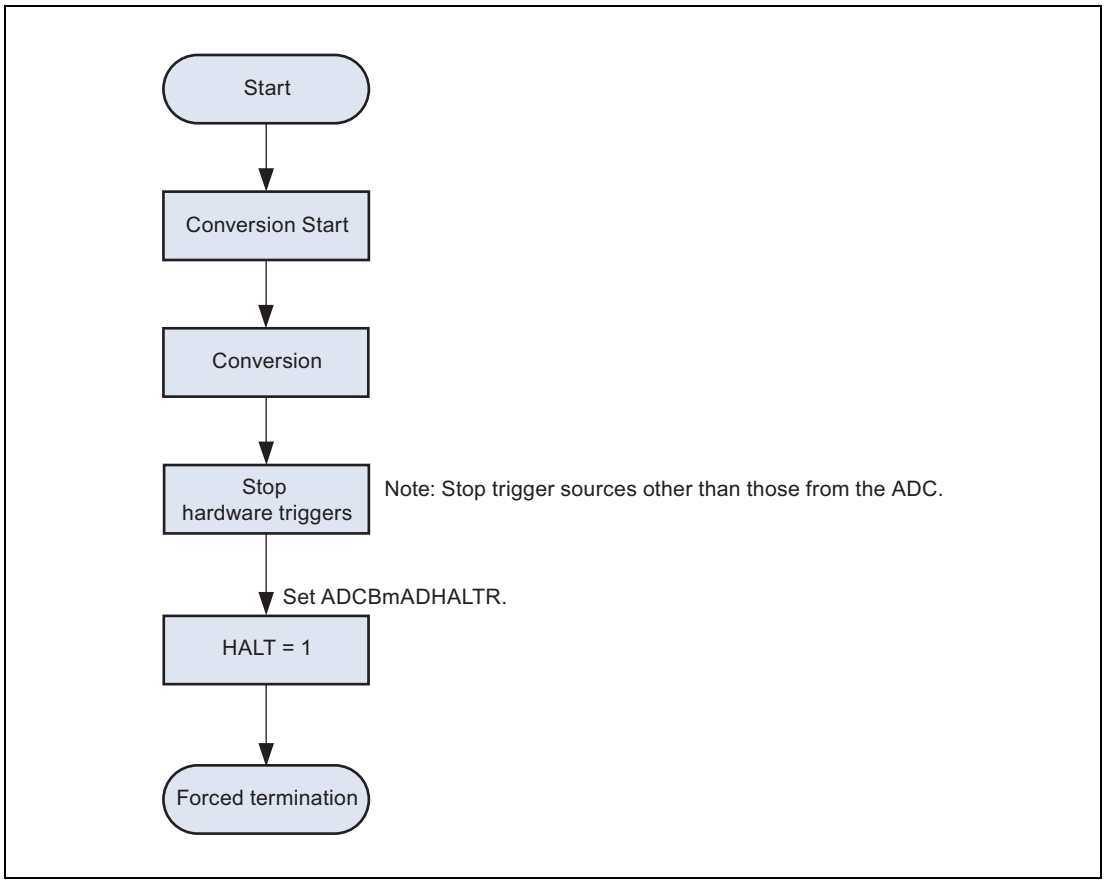


Figure 26.6 Terminating Procedure

NOTE

m = 0, 1

26.7.2 Examples of Normal A/D Conversion Operation

26.7.2.1 Multicycle Scan Mode

The following figure shows an example of operation when converting four virtual channels by 2-cycle scan for scan group 0 in multicycle scan mode by using normal A/D conversion mode (CNVCLS[2:0] = 0_H).

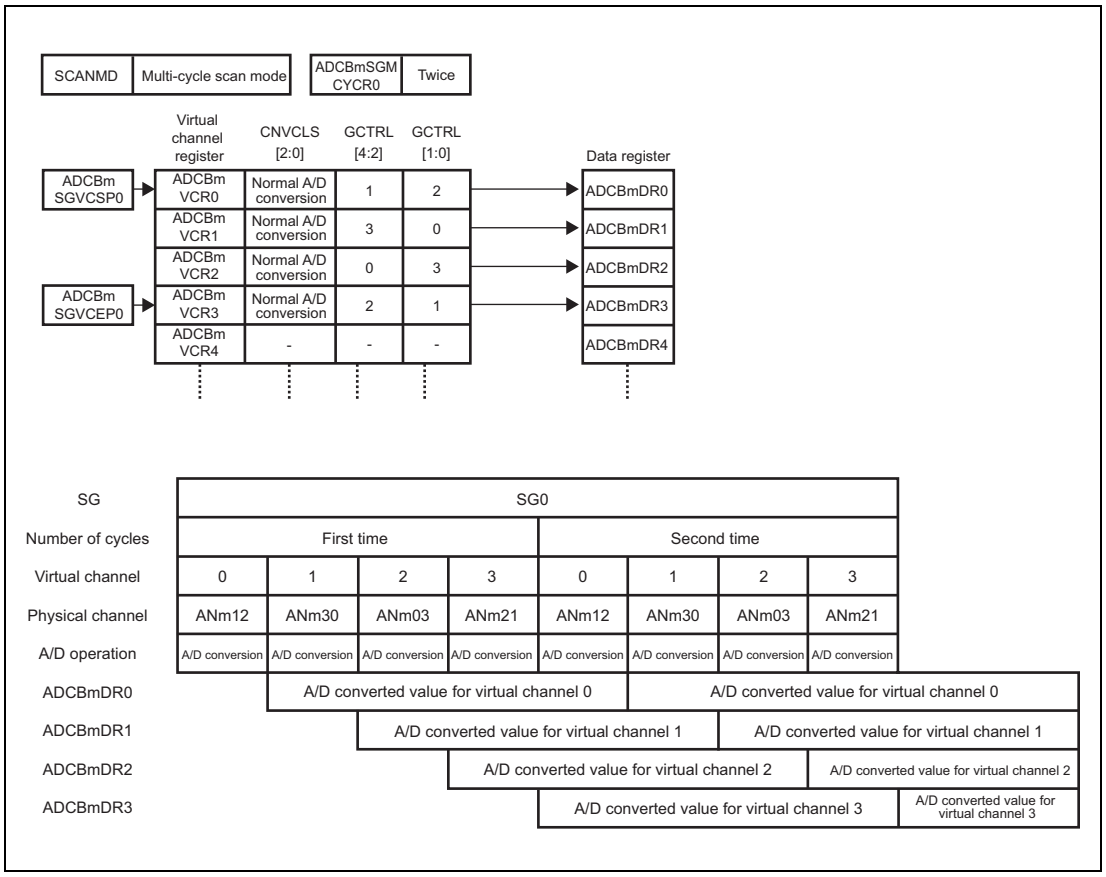


Figure 26.7 Example of Operation in Multicycle Scan Mode

NOTE

m = 0, 1

26.7.2.2 Continuous Scan Mode

Figure 26.8 shows an example of operation when converting four virtual channels for scan group 0 in continuous scan mode by using normal A/D conversion mode (CNVCLS[2:0] = 0_H).

In this mode, if a trigger of a lower-priority scan group is input to a scan group for which continuous scan mode is set (SCANMD = 1_H), the trigger is not accepted. Therefore, it is assumed that continuous scan mode is set for scan group 0.

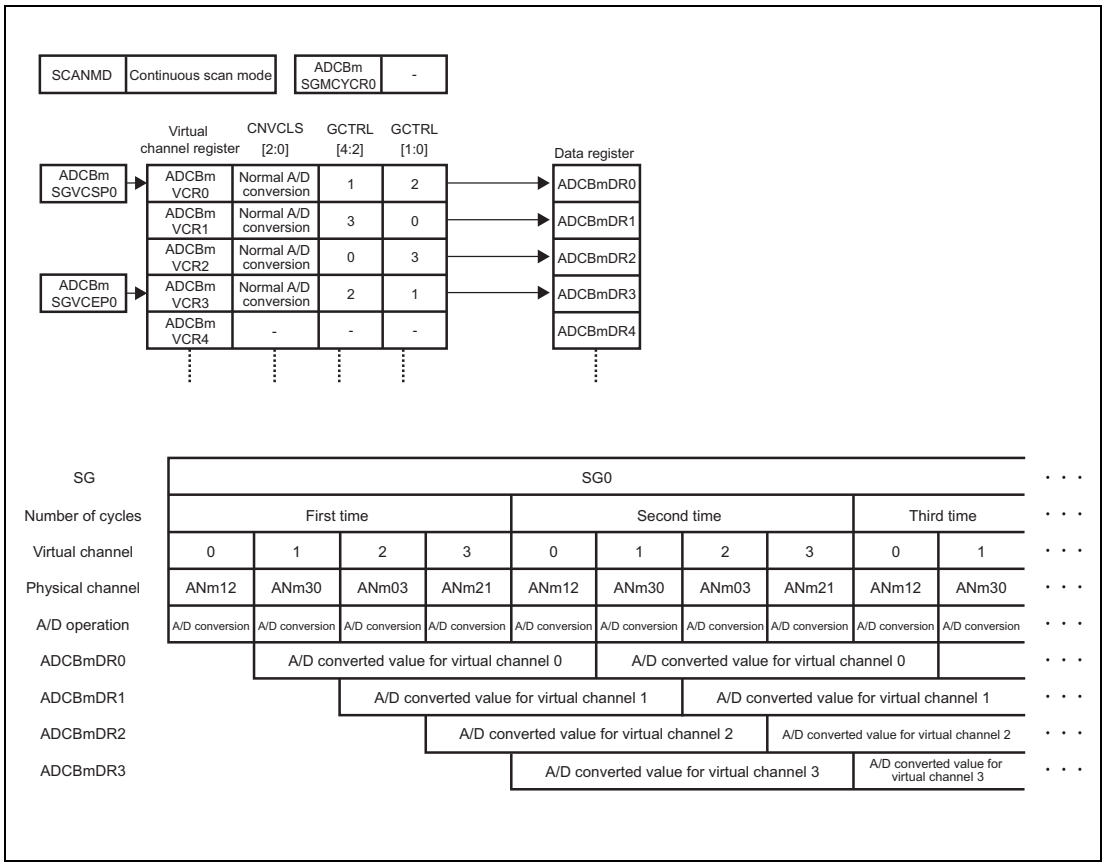


Figure 26.8 Example of Operation in Continuous Scan Mode

NOTE

m = 0, 1

26.7.3 Example of Simultaneous Track-and-Hold Operation

26.7.3.1 Simultaneous Track-and-Hold Operation (Physical Channel Group Selected)

Figure 26.9 shows an example of simultaneous track-and-hold operation when execution of selected T&H (CNVCLS[2:0] = 2_H with physical channel group selected), A/D conversion of the hold value (CNVCLS[2:0] = 1_H), and self-diagnosis (CNVCLS[2:0] = 3_H) are performed.

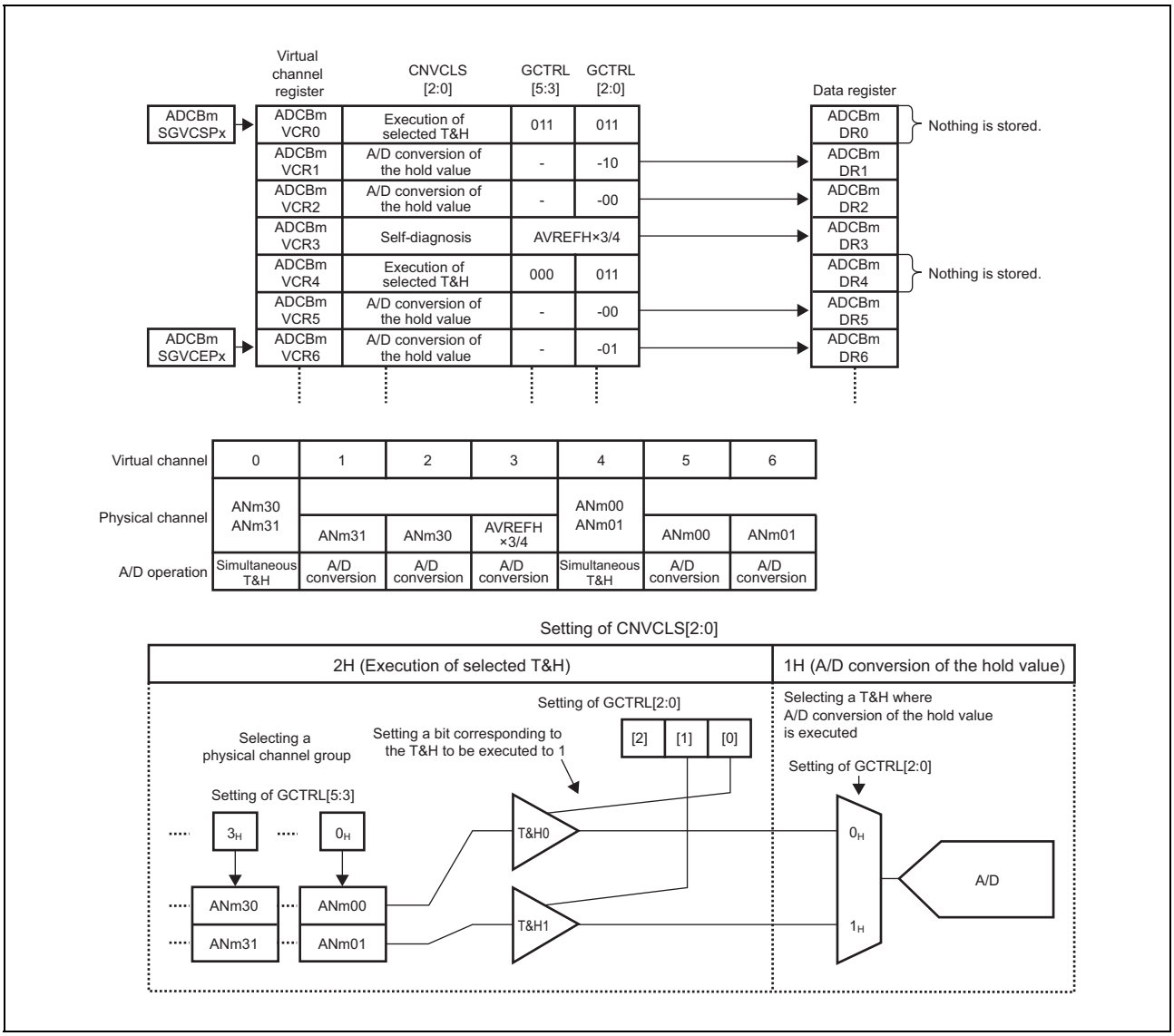


Figure 26.9 Example of Simultaneous Track-and-Hold Operation (Physical Channel Group Selected)

NOTES

m = 0; x = 0 to 4

26.7.4 Example of Normal A/D Conversion Operation in Addition Mode

Figure 26.10 shows an example of operation when converting four virtual channels for scan group 0 by using normal A/D conversion (CLVCLS[2:0] = 4_H) in addition mode.

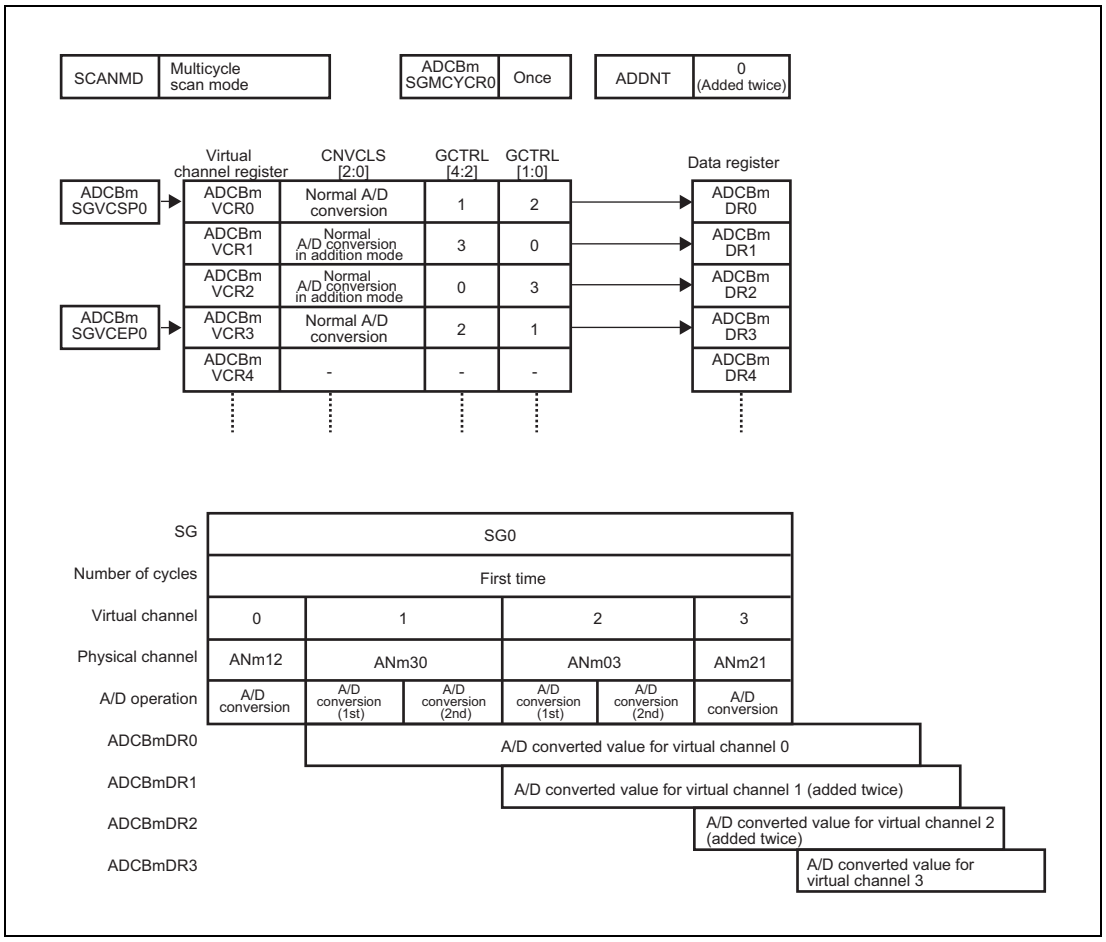


Figure 26.10 Example of Normal A/D Conversion Operation in Addition Mode

NOTE

m = 0, 1

26.7.5 Example of Operation of External Analog Multiplexer

The following shows examples of operation of an external analog multiplexer using normal A/D conversion with the MPX mode (CNVCLS[2:0] = 5_H) or normal A/D conversion in addition mode with the MPX mode (CNVCLS[2:0] = 6_H).

26.7.5.1 Example of Using an External Analog Multiplexer (Port Output)

Figure 26.11 shows an example of port output using an external analog multiplexer.

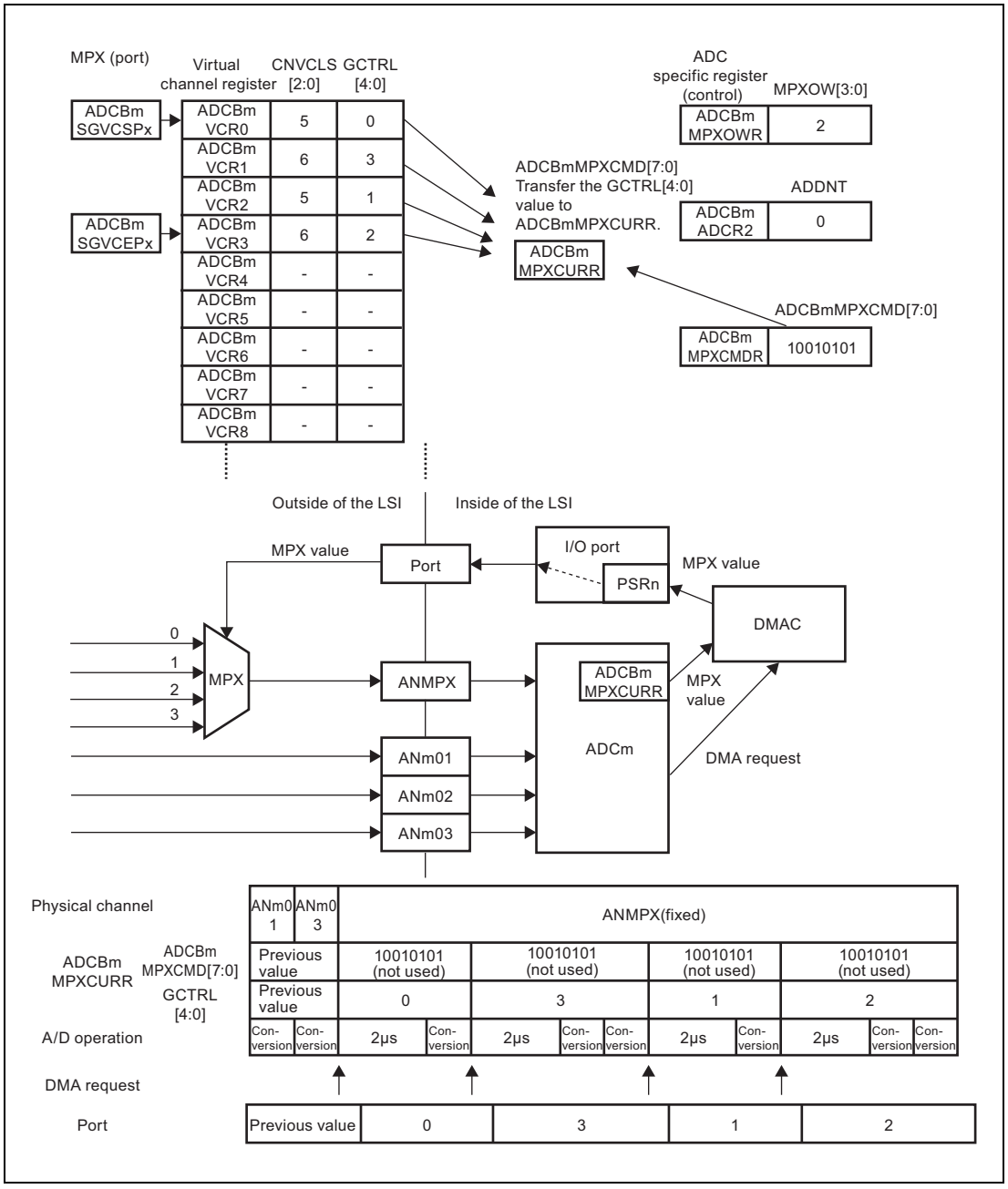


Figure 26.11 Example of Using an External Analog Multiplexer (Port Output)

NOTE

m = 0: ANMPX = AN043; m = 1: ANMPX = AN100; x = 0 to 4

26.7.5.2 Example of Using an External Analog Multiplexer (SPI Output)

Figure 26.12 shows an example of SPI output using an external analog multiplexer.

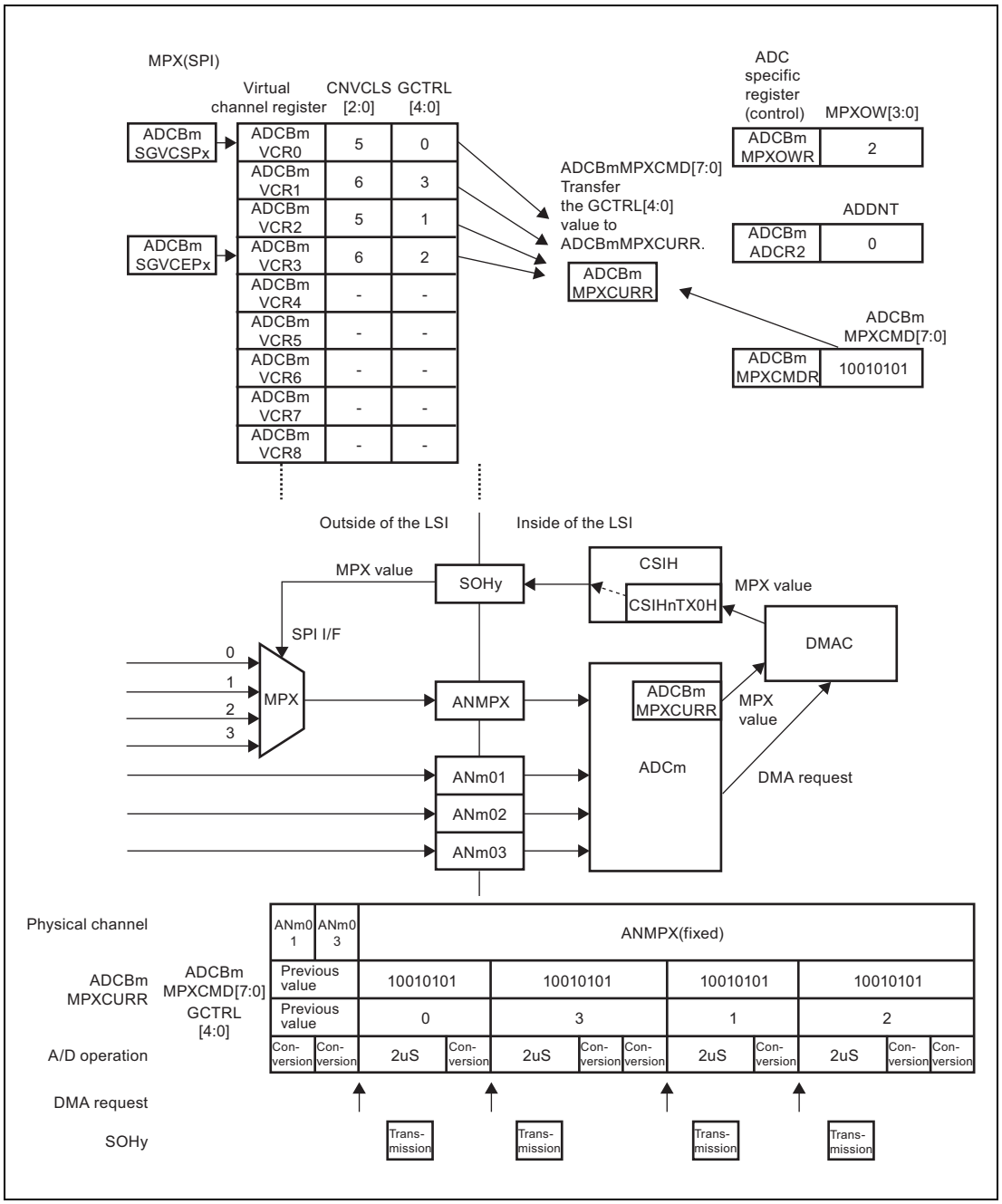


Figure 26.12 Example of Using an External Analog Multiplexer (SPI Output)

NOTE

m = 0: ANMPX = AN043; m = 1: ANMPX = AN100; x = 0 to 4

26.7.6 Example of Synchronous Suspend and Resume Operation

Figure 26.13 shows an example of synchronous suspend and resume operation when a higher-priority SG interrupts a lower-priority SG.

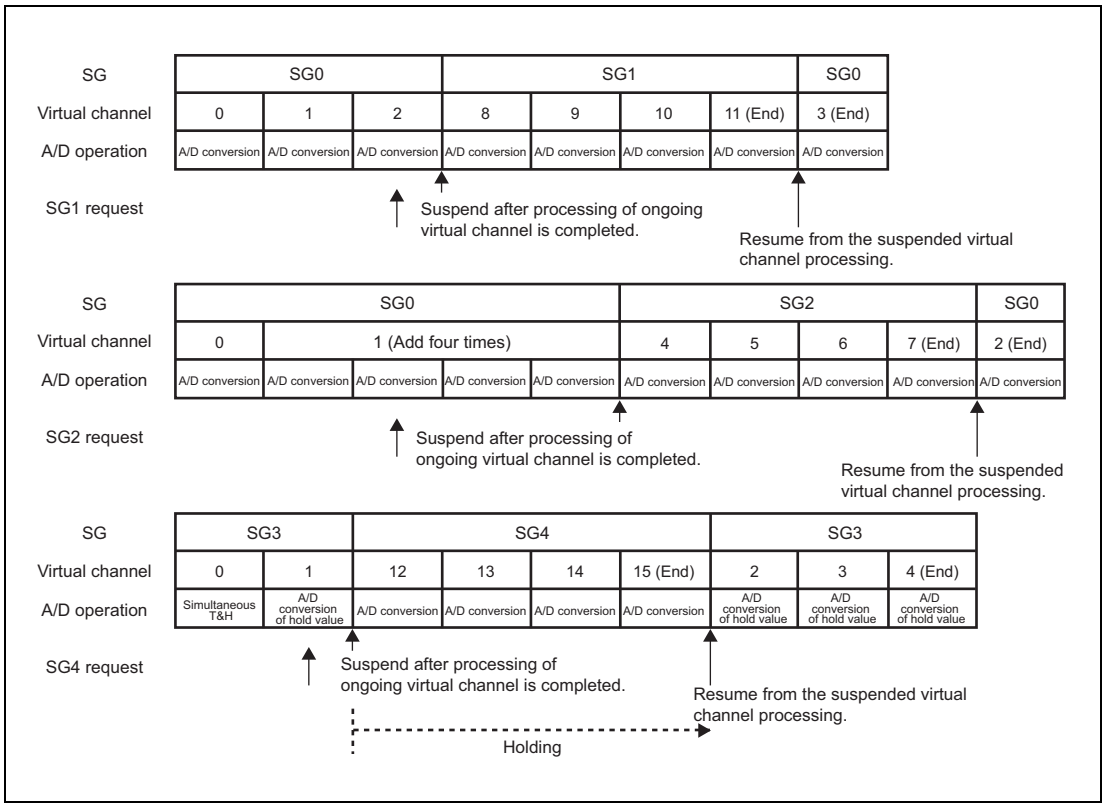


Figure 26.13 Example of Synchronous Suspend and Resume Operation

CAUTION

Priority of scan groups:

Low High

SG0 < SG1 < SG2 < SG3 < SG4

26.7.7 Example of Asynchronous Suspend and Resume Operation

Figure 26.14 shows an example of asynchronous suspend and resume operation when a higher-priority SG interrupts a lower-priority SG.

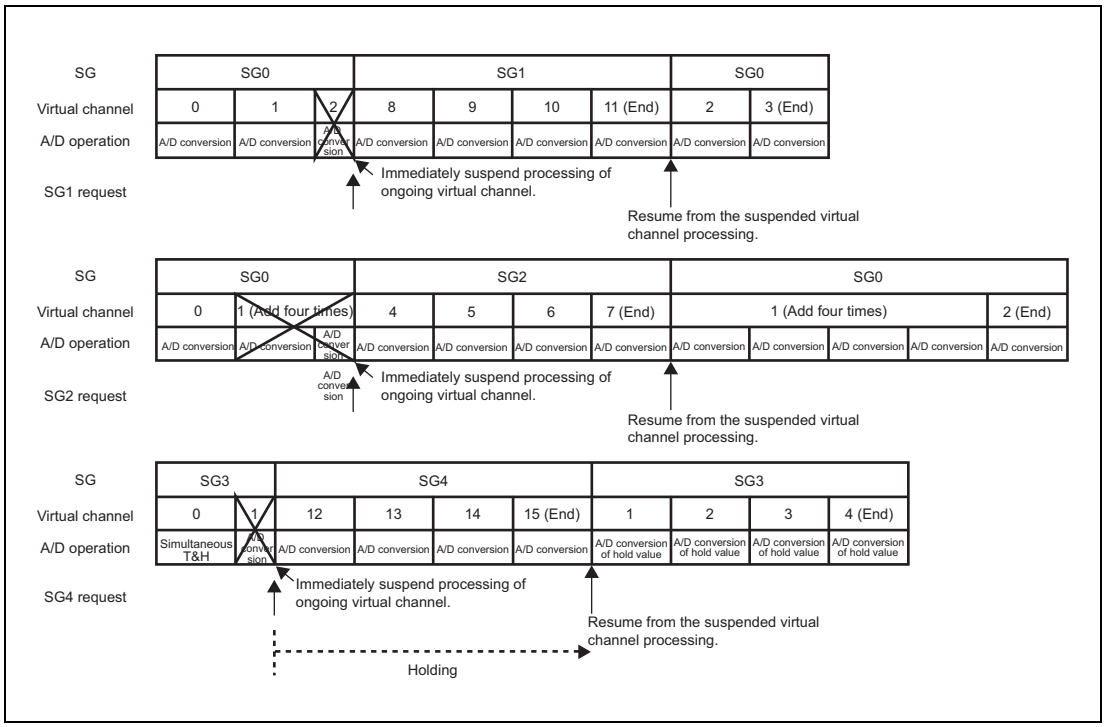


Figure 26.14 Example of Asynchronous Suspend and Resume Operation

CAUTION

Priority of scan groups:

Low High

SG0 < SG1 < SG2 < SG3 < SG4

26.7.8 Example of A/D Timer Operation

Figure 26.15 shows an example of A/D timer operation.

The A/D timer counts in synchronization of the 40-MHz clock.

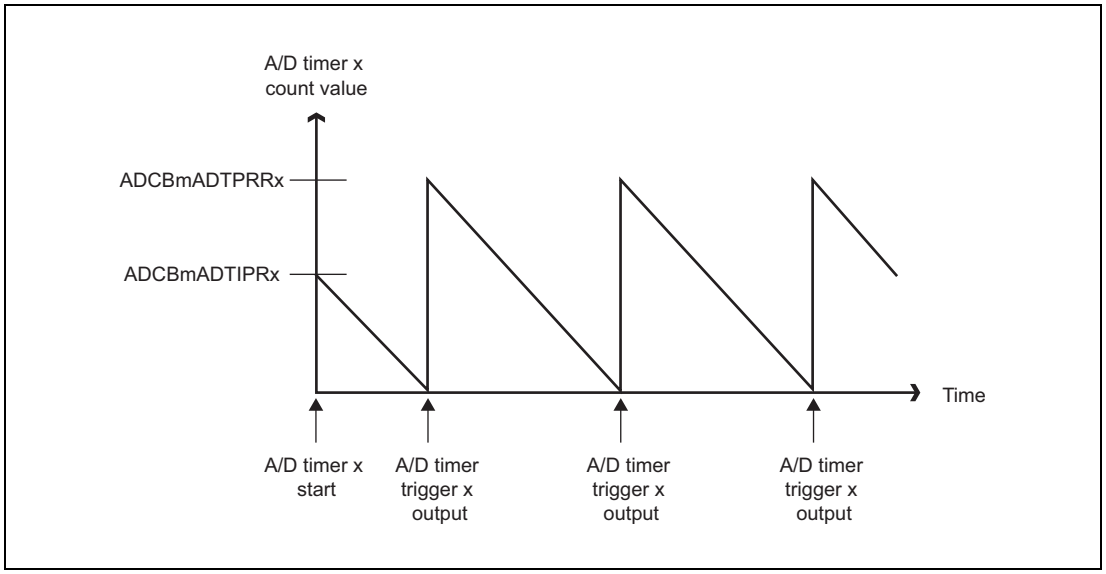


Figure 26.15 Example of A/D Timer Operation

NOTE

m = 0, 1; x = 3, 4

26.7.9 Self-Diagnostic Functions

The ADC is equipped with the following three self-diagnostic functions.

- Pin-level self-diagnosis
- A/D conversion circuit self-diagnosis
- Wiring-break detection self-diagnosis

26.7.9.1 Pin-Level Self-Diagnosis

The pin-level self-diagnosis performs A/D conversion that is set to a different voltage for even-number physical channel groups and odd-number physical channel groups to check an abnormal path from the ANI.

The different voltage setting is made in ADCBmTDCR and can be detected by combination of AVSS, AVCC, and 1/2 × AVCC.

Features and settings of the pin-level self-diagnosis of the ADC are described below.

[Features]

1. Users can select desired physical channels to be tested.
2. AVSS, AVCC, or 1/2AVCC is selectable as a self-diagnosis level.
3. Performing A/D conversion for SG0 to SG4 makes the pin-level self-diagnosis available.

[Settings]

1. Make settings according to the initial settings (**Figure 26.3, Initial Settings**).
2. Set CNVCLS[2:0] in the virtual channel register ADCBmVCRn to 0_H and set GCTRL[4:0] to select desired channels.
3. Set TDE in the pin-level self-diagnosis control register to 1_H and set TDLV[1:0] to specify the desired level of pin-level self-diagnosis.
4. Make other settings required for A/D conversion according to the initial settings (**Figure 26.3, Initial Settings**).
5. Assert SG0 to SG4 trigger signals to perform A/D conversion.

26.7.9.2 A/D Converter Self-Diagnosis

The A/D converter self-diagnosis is used to verify that A/D conversion operates correctly.

The voltage value setting is made in GCTRL[4:0] when CNVCLS[2:0] = 3_H, and can be converted for AVREFH × 1, AVREFH × 3/4, AVREFH × 1/2, AVREFH × 1/4, and AVREFH × 0.

Features and settings of the A/D conversion circuit self-diagnosis are described below.

[Features]

1. AVREFH × 1, AVREFH × 3/4, AVREFH × 1/2, AVREFH × 1/4, or AVREFH × 0 is selectable as a self-diagnosis voltage level.
2. Performing A/D conversion for SG0 to SG4 makes the A/D converter self-diagnosis available.

[Settings]

1. Make settings according to the initial settings (**Figure 26.3, Initial Settings**).
2. Set CNVCLS[2:0] in the virtual channel register ADCBmVCRn to 3_H and set GCTRL[5:0] to specify the desired self-diagnosis voltage level.
3. Make other settings required for A/D conversion according to the initial settings (**Figure 26.3, Initial Settings**).
4. Assert SG0 to SG4 trigger signals to perform A/D conversion.

NOTE

m = 0, 1; n = 0 to 39

Applying A/D converter self-diagnosis to a pin which has handled A/D conversion before A/D converter self-diagnosis while an injection current is flowing may affect the accuracy of the result of conversion. Accordingly, select pins through which a current will not be being injected as targets for A/D converter self-diagnosis.

Example settings are as follows.

CAUTION

Applying a voltage above the power supply voltage or below the ground voltage to a pin will lead to an injection current flowing.

(1) If A/D conversion and A/D converter self-diagnosis proceed in the same SG, select pins through which a current will not be being injected before A/D converter self-diagnosis.

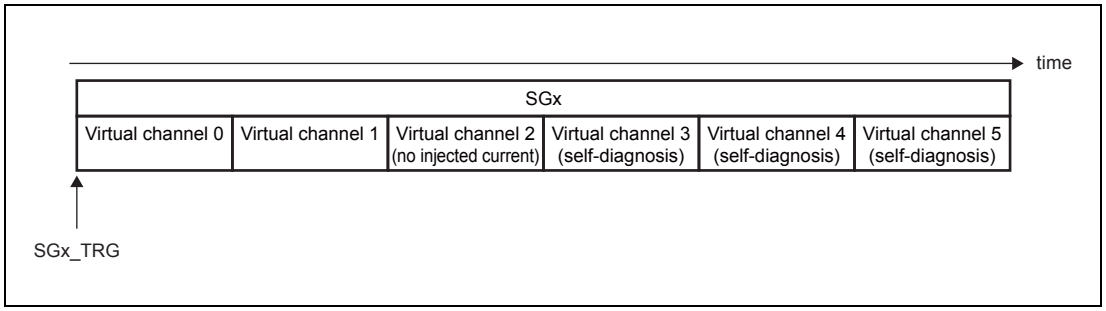


Figure 26.16 Example Setting when A/D Conversion and A/D Converter Diagnosis Proceed in the Same SG

(2) If a scan group has higher priority than a scan group to which A/D converter self-diagnosis is to be applied, the self-diagnosis may be suspended to allow A/D conversion by the higher-priority SG to proceed, and then resumed. If this is the case, select pins*1 through which current will not be being injected at the end of conversion by the high-priority SG.

Note 1. If you cannot determine which pins will not have current being injected, proceed with A/D conversion on an unused pin as an alternative.

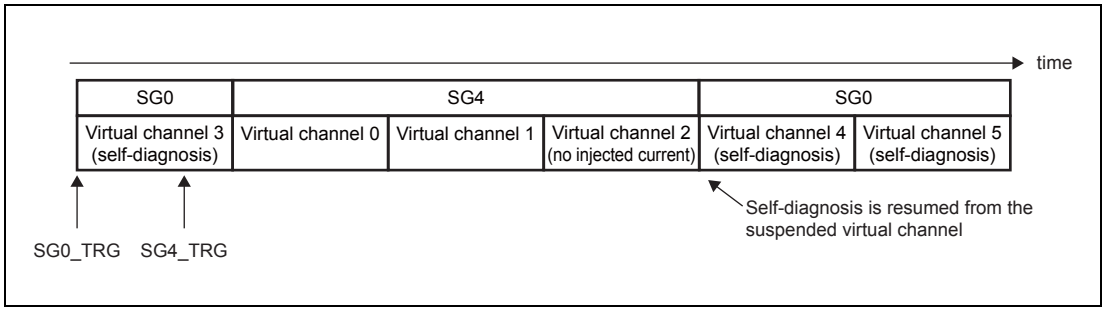


Figure 26.17 Example Setting when a Scan Group has Higher Priority than a Scan Group to which A/D Converter Self-Diagnosis is to be Applied

26.7.9.3 Wiring-Break Detection Self-Diagnosis

The wiring-break detection self-diagnosis detects wiring-break of the ANI. If a wiring-break is present, the conversion result attenuates to approximately 0 V, and an abnormal value is detected in the conversion result. This can be determined as detection of wiring-break.

[Feature]

1. Users can select desired physical channels for which wiring-break is to be detected.

[Settings]

1. Make settings according to the initial settings (**Figure 26.3, Initial Settings**).
2. Set CNVCLS[2:0] in the virtual channel register ADCBmVCRn to 0_H and set GCTRL[5:0] to select desired channels.
3. Set ODE in the wiring-break detection control register to 1_H and set ODPW[5:0] to specify the desired wiring-break detection pulse width.
4. Assert SG0 to SG4 trigger signals to perform A/D conversion.

NOTE

m = 0, 1; n = 0 to 39

26.7.10 Sampling of an Analog Input and Processing Time for a Scan Group

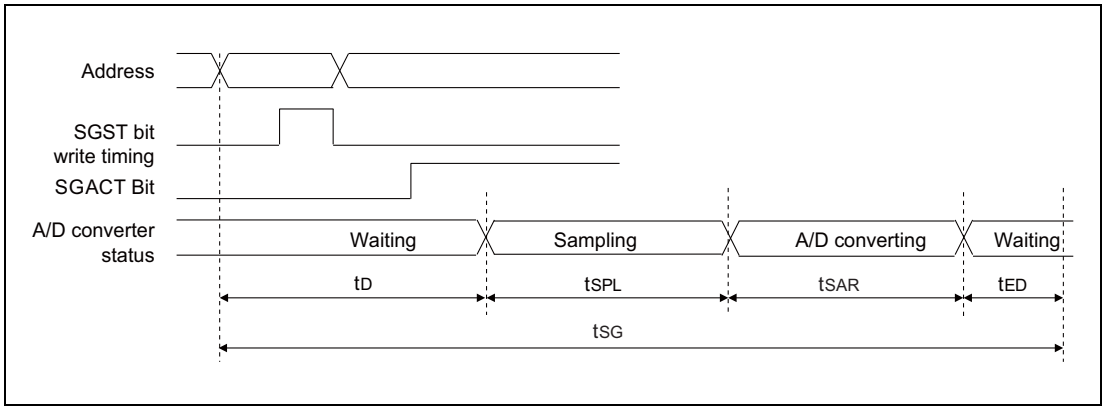


Figure 26.18 Timing Chart of Normal A/D Conversion Operation (Single Conversion)

The track-and-hold circuit is equipped subsequent to the analog input. The ADC has an internal sample-and-hold circuit. After the SGST bit in ADCBmSGCRx is set to 1, the ADC performs sampling after the scan group start delay time (tD) has passed, and then starts successive approximation conversion processing.

Figure 26.18, Timing Chart of Normal A/D Conversion Operation (Single Conversion) shows the operation timing in one-cycle scan for a single virtual channel. The processing time for a scan group (tSG) includes the scan group start delay time (tD), the sampling time (tSPL), the successive approximation conversion processing time (tSAR), and the scan group end delay time (tED). **Table 26.40, Processing Time for a Scan Group** lists the processing time for a scan group.

The processing time for a scan group (tSG) can be obtained from the following formula, where the number of virtual channels is i and the number of multicycles is j in multicycle scan mode:

$$tSG = tD + (tSPL + tSAR) \times i \times j + tED$$

The processing time for the first cycle of scan in continuous scan mode is as follows:

$$tD + (tSPL + tSAR) \times i$$

The processing time for the second (or subsequent) cycle of scan in continuous scan mode is as follows:

$$(tSPL + tSAR) \times i$$

Table 26.40 Processing Time for a Scan Group

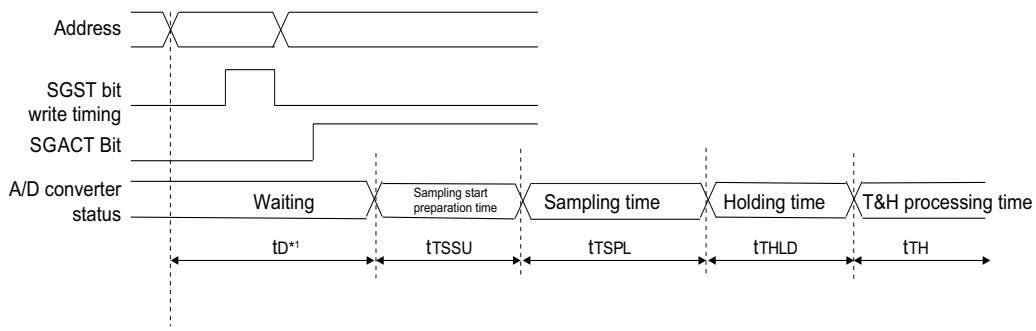
Item	Symbol	Period	Unit
Scan group start delay time	tD	(2 to 4) × Pϕ + 5 × Iϕ	Pϕ (P-Bus clock) Iϕ (Internal clock)
Sampling time	tSPL	18 × Iϕ	Iϕ (Internal clock)
Successive approximation conversion processing time	tSAR	22 × Iϕ	Iϕ (Internal clock)
Scan group end delay time	tED	(2 to 4) × Iϕ + 3 × Pϕ	Pϕ (P-Bus clock) Iϕ (Internal clock)
Scan group processing time	tSG	47 × Iϕ + 5 × Pϕ to 49 × Iϕ + 7 × Pϕ	Pϕ (P-Bus clock) Iϕ (Internal clock)

NOTE

m = 0, 1; x = 0 to 4

"Pϕ" and "Iϕ" in the table respectively represent the low-speed peripheral clock (CLK_LSB) and the non-modulated low-speed peripheral clock (CLKC_LSB).

26.7.10.1 Processing Time for Execution of Selected T&H and A/D Conversion of Hold Value



Note 1. For tD, see **Figure 26.18**.

Figure 26.19 Timing Chart for Operation of Execution of Selected T&H and A/D Conversion of Hold Value

The track-and-hold circuit is equipped subsequent to the analog input. **Table 26.41** shows the specified time period from sampling start to transition to the hold state (tTSPL) and the specified time period from transition to the hold state to A/D conversion start (tTHLD), and also shows the T&H processing time (tTH) when A/D conversion of the hold value is performed in a virtual channel.

Table 26.41 Track-and-Hold Processing Time

Item	Symbol	Period	Unit
Sampling start preparation time	tTSSSU	3	Iϕ (internal clock)
Sampling time	tTSPL	27	Iϕ (internal clock)
Hold time	tTHLD	10	Iϕ (internal clock)
T&H processing time	tTH	40	Iϕ (internal clock)

26.7.11 Selection of Trigger Input for a Scan Group

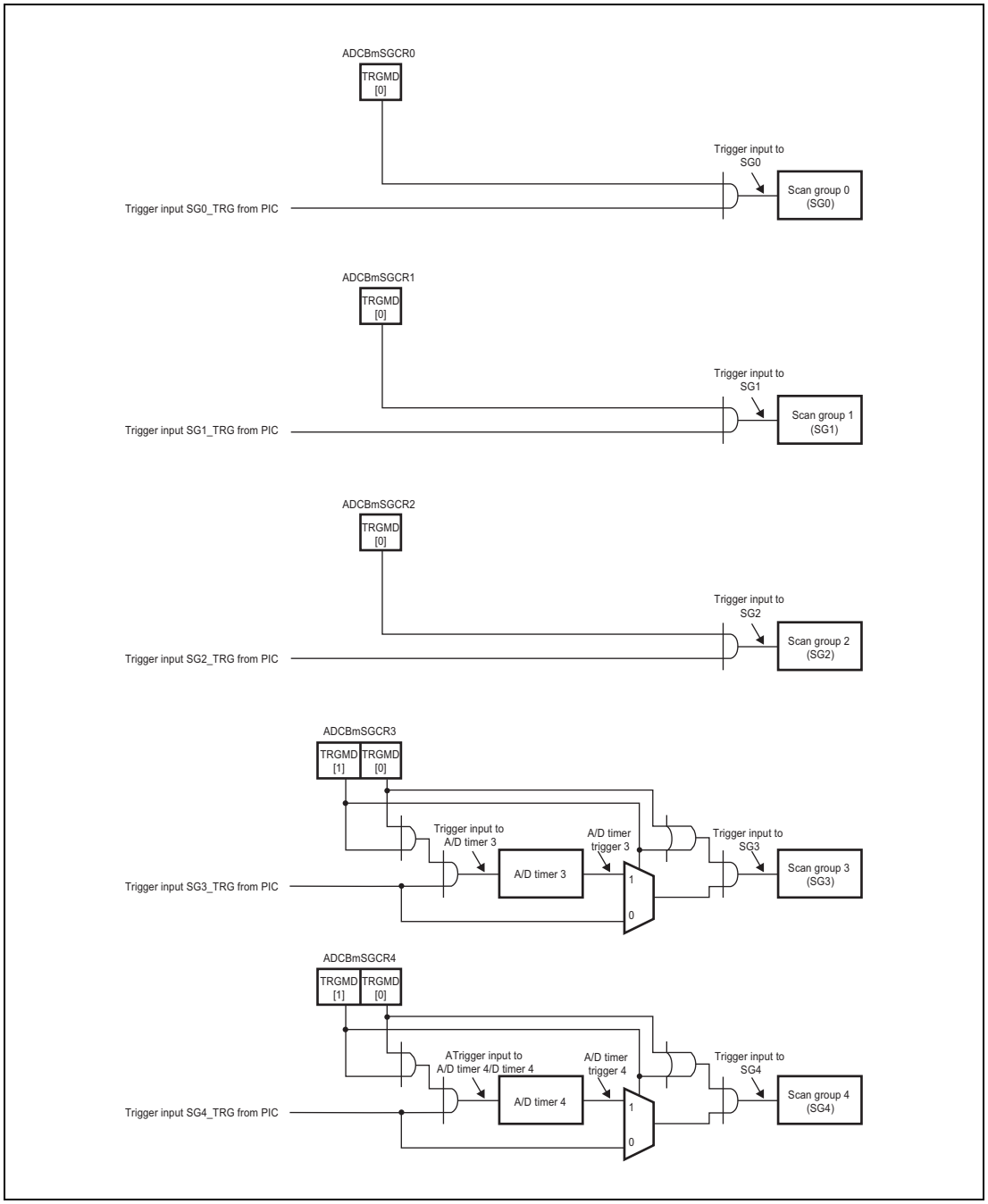


Figure 26.20 Selection of Trigger Input for a Scan Group

NOTE

m = 0, 1

26.7.12 Starting a Scan Group by Using a Hardware Trigger

Scan group x can be started by using an input from hardware trigger SGx_TRG. To start scan group x by an input from hardware trigger SGx_TRG, set TRGMD in ADCBmSGCRx to 1_H. When the selected hardware trigger SGx_TRG is input in this state, SGACT is set to 1. The timing since SGACT is set to 1 until scan group x is started is the same as the timing when SGST is set to 1 by a software trigger.

NOTE

m = 0, 1; x = 0 to 4

26.7.13 Starting a Scan Group by Using an A/D Timer Trigger

Scan group 3 or 4 can be started by using a trigger from A/D timer 3 or 4. To start scan group 3 or 4 by using a trigger from A/D timer 3 or 4, set TRGMD in ADCBmSGCR3 or ADCBmSGCR4 to 2_H. Furthermore, set ADTST of scan group 3 or 4 to 1 to start A/D timer 3 or 4.

When a timer trigger is input in this state, SGACT is set to 1. The timing since SGACT is set to 1 until scan group 3 or 4 is started is the same as the timing when SGST is set to 1 by a software trigger.

NOTE

m = 0, 1

26.7.14 Starting A/D Timer by Using a Hardware Trigger

A/D timer 3 or 4 can be started by using a hardware trigger SG3_TRG or SG4_TRG input. To start A/D timer 3 or 4 by using a hardware trigger SG3_TRG or SG4_TRG input, set TRGMD in ADCBmSGCR3 or ADCBmSGCR4 to 3_H. When the selected external trigger is input in this state, A/D timer 3 or 4 starts. Furthermore, a trigger from A/D timer 3 or 4 starts scan group 3 or 4.

NOTE

m = 0, 1

26.7.15 Monitoring Function by Using the A/D Conversion Monitor Pin

ADENDm can be used to monitor the processing timing of the virtual channel specified by ADCBmADENDP. For details of pin settings, see **Section 2, Pins**.

Figure 26.21 shows the A/D conversion monitor timing.

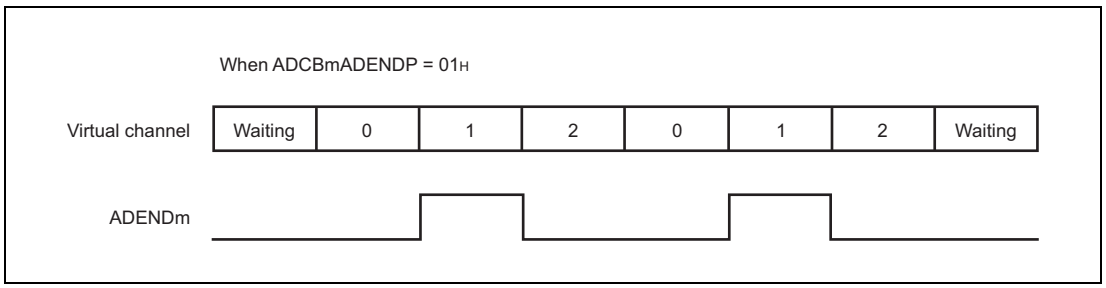


Figure 26.21 A/D Conversion Monitor Timing

CAUTION

If the high-level voltage is output from ADENDm in a lower-priority scan group and a higher-priority scan group suspends (asynchronous suspend) the processing of the lower-priority scan group, the low-level voltage is output from ADENDm. Since the suspended virtual channel processing for the lower-priority scan group resumes after that, the high-level voltage is output again from ADENDm.

NOTE

m = 0, 1

26.7.16 Scan End Interrupt Request

Scan group x can issue a scan end interrupt request (ADImx) to the INTC. When ADIE in ADCBmSGCRx is set to 1, ADImx can be output after the SGx scan ends. When ADIE in ADCBmSGCRx is set to 0, the ADImx output at the end of the SGx scan can be disabled. When ADIE in ADCBmVCRn is set to 1, ADImx can be output when A/D conversion for virtual channel n in SGx ends. When ADIE in ADCBmVCRn is set to 0, the ADImx output at the end of the A/D conversion for virtual channel n in SGx can be disabled. The setting of ADIE in ADCBmSGCRx is independent of the setting of ADIE in ADCBmVCRn.

Example 1) A scan is executed for virtual channel 0 or 1 in SG0 when ADIE in ADCBmSGCR0 is 0, ADIE in ADCBmVCR0 is 1, and ADIE in ADCBmVCR1 is 0.
ADIm0 is output when A/D conversion ends for virtual channel 0.

Example 2) A scan is executed for virtual channel 0 or 1 in SG0 when ADIE in ADCBmSGCR0 is 0, ADIE in ADCBmVCR0 is 1, and ADIE in ADCBmVCR1 is 1.
ADIm0 is output when A/D conversion ends for virtual channel 0 and virtual channel 1.

Example 3) A scan is executed for virtual channel 0 or 1 in SG0 when ADIE in ADCBmSGCR0 is 1, ADIE in ADCBmVCR0 is 0, and ADIE in ADCBmVCR1 is 0.
ADIm0 is output when a scan ends (at the end of A/D conversion for virtual channel 1).
ADIm0 is output at each two cycles (or more) in multicycle scan mode, or each time a scan ends (at the end of A/D conversion for virtual channel 1) in continuous scan mode.

Furthermore, the DMAC can be activated when ADImx occurs.

For the DMAC setting, see **Section 7, DMA**.

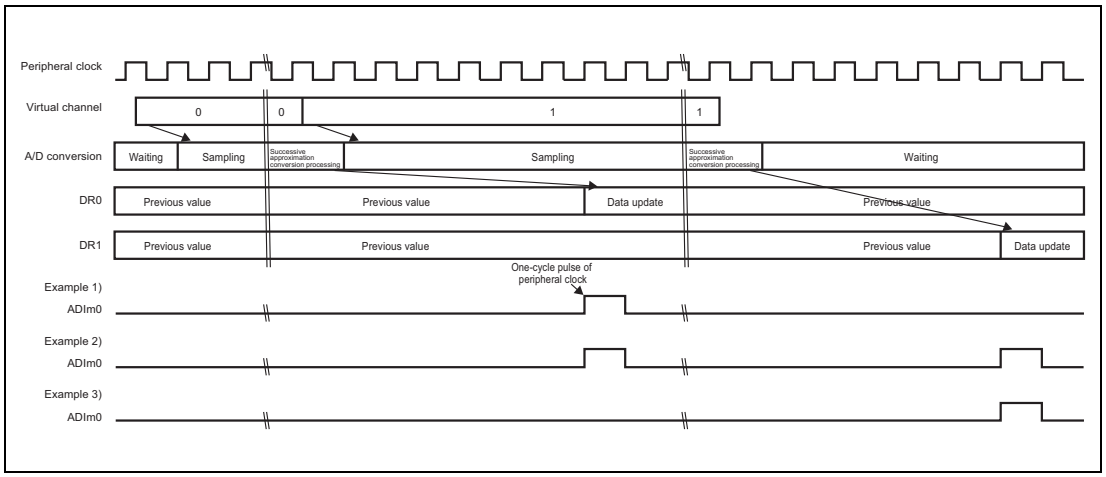


Figure 26.22 Scan Conversion End Interrupt Occurrence Timing

NOTE

m = 0, 1; x = 0 to 4; n = 0 to 39

26.7.17 MPX Interrupt Request

The ADC can issue an MPX interrupt request (ADMPXIm) to the INTC. ADMPXIm is generated when a virtual channel for which CNVCLS[2:0] in ADCBmVCRn is set to 5_H or 6_H is started.

The DMAC can be activated when ADMPXIm occurs.

For the DMAC setting, see **Section 7, DMA**.

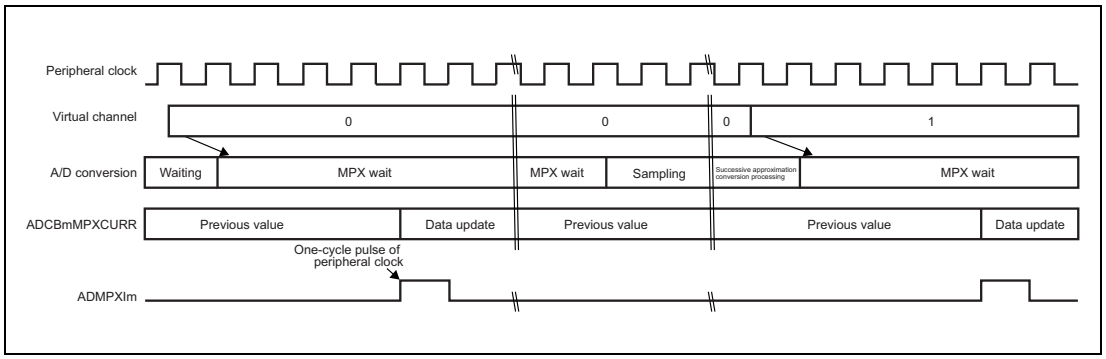


Figure 26.23 Example of an MPX Interrupt Occurrence

NOTE

m = 0, 1; n = 0 to 39

26.7.18 A/D Error Interrupt Request and A/D Parity Error Notification

The ADC can issue an A/D error interrupt request (ADEm) to the INTC and an A/D parity error notification (ADPEm) to the error control module (ECM). For an error source for which ULEIE, OWEIE, and IDEIE in ADCBmSFTCR is set to 1, the OR condition of the error source is issued as ADEm. For an error source for which ULEIE, OWEIE, and IDEIE in ADCBmSFTCR is set to 0, ADEm can be disabled. ADPEm is enabled when PEIE in ADCBmSFTCR is set to 1. ADPEm is disabled when PEIE in ADCBmSFTCR is set to 0.

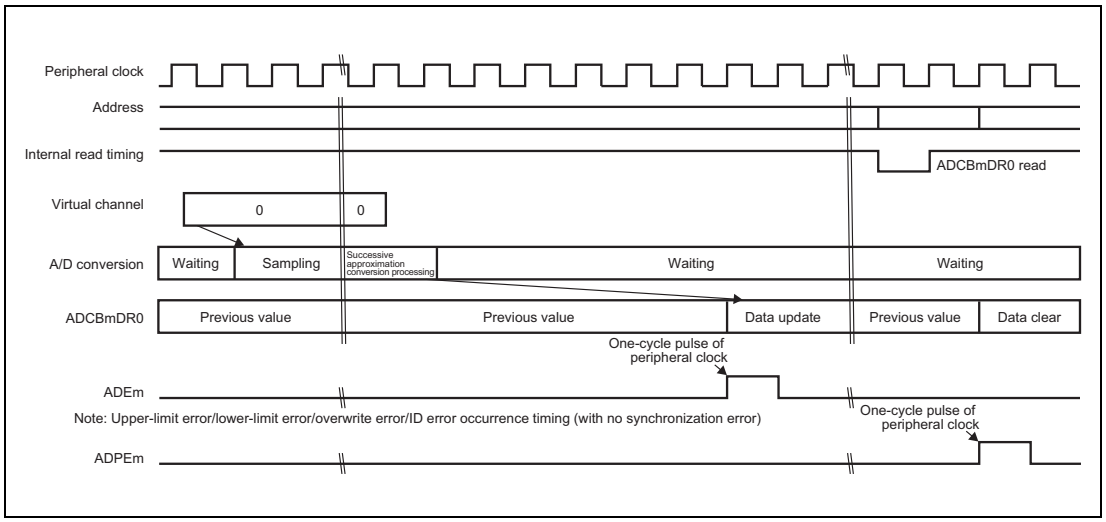


Figure 26.24 Example of an Occurrence of A/D Error Interrupt and A/D Parity Error Notification

NOTES

m = 0, 1

26.7.19 DFE/ASF Entry Function

The ADC can issue a request for entry to the DFE and a request for entry to the ASF for both DFE and ASF to enable entry according to the settings of DFENT in ADCBmVCRn and DFENTSGxE and ASENTSGxE in ADCBmDFASENTSGER. At the same time, the ADC outputs the TAG information and A/D conversion data that are set in DFTAG in ADCBmVCRn. For the format of the output A/D conversion data, see **Section 26.5.2, ADCBmDRn — Data Register n**.

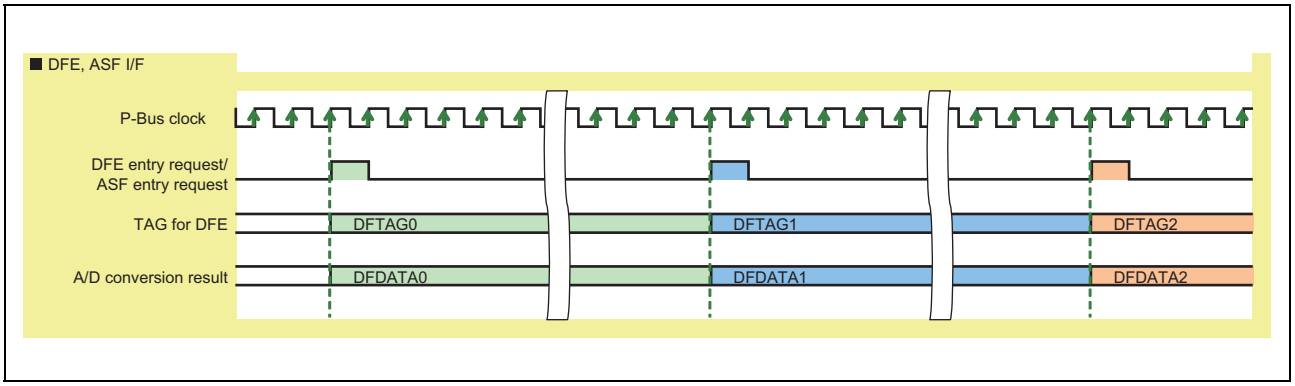


Figure 26.25 DFE/ASF Entry Timing

NOTE

m = 0, 1; n = 0 to 39; x = 0 to 4

26.8 Definition of A/D Conversion Accuracy

A/D conversion accuracy is defined below.

- Resolution
  Number of digital output codes of the A/D converter
- Quantization error
  An error essentially contained in A/D converters, which is given as 1/2LSB (**Figure 26.26**).
- Offset error
  Deviation of the analog input voltage value from the ideal A/D conversion characteristics when the digital output changes from the minimum voltage value 000_H to 001_H. However, the quantization error is not included (**Figure 26.26**).
- Full-scale error
  Deviation of the analog input voltage value from the ideal A/D conversion characteristics when the digital output changes from FFE_H to FFF_H. However, the quantization error is not included (**Figure 26.26**).
- DNL (Differential nonlinear error)
  Deviation between the ideal digital output code width (Vq) and the actual digital output code width (Va), which is given as (Va - Vq)/Vq. However, the offset error, the full-scale error, and the quantization error are not included (**Figure 26.26**).
- INL (Integral nonlinear error)
  Deviation of the actual value from the ideal A/D conversion characteristics, from the zero voltage to the full-scale voltage, which is given as an integral of DNL from 000_H to a digital output code. However, the offset error, the full-scale error, and the quantization error are not included (**Figure 26.26**).
- Absolute accuracy
  Deviation between the digital value and the analog input value. The offset error, the full-scale error, the quantization error, DNL, and INL are included (**Figure 26.26**).

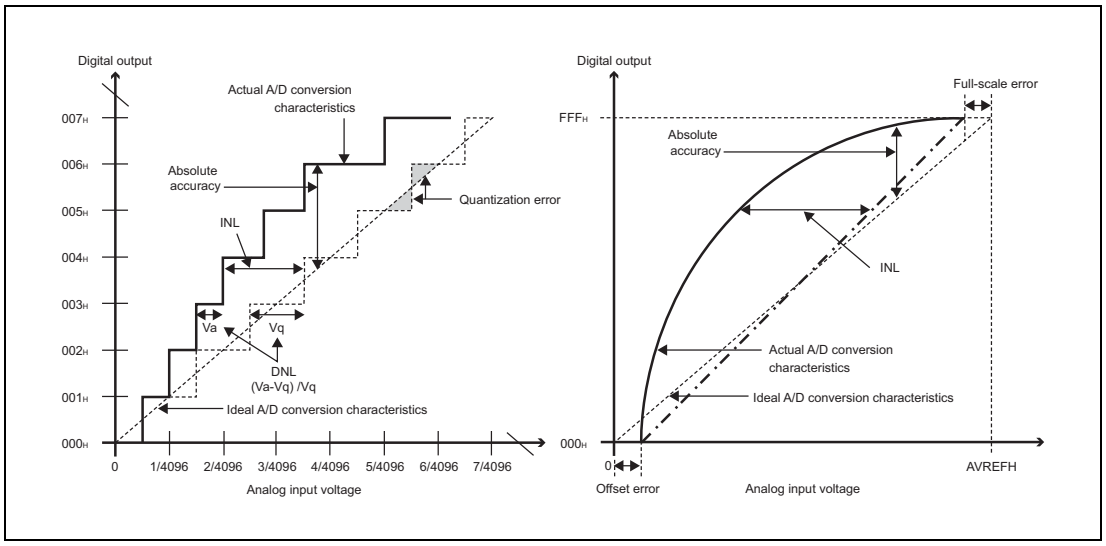


Figure 26.26 Definition of A/D Conversion Accuracy

26.9 Usage Notes

26.9.1 Notes on Using an External Analog Multiplexer

When you use an external multiplexer, observe the following notes so as not to cause system crash.

Except for the exceptions below, set the MPX wait as follows:

- When an MPX value is transferred at a port: Insert a wait of at least 1 µs.
- When an MPX value is transferred with the SPI interface: Insert a wait of at least SPI transmission time + 1 µs.

Exception 1) When an external multiplexer is used for one scan group

When an external multiplexer is used for SG0 with SUSMTD[1:0] set to 1_H, or when an external multiplexer is used for any of SG0 to SG3 with SUSMTD[1:0] set to 2_H, set the MPX wait as follows:

- When an MPX value is transferred at a port: Insert a wait of at least 1 µs.
- When an MPX value is transferred with the SPI interface: Insert a wait of at least SPI transmission time × 2.

Exception 2) When an external multiplexer is used for multiple scan groups

When an external multiplexer is used with SUSMTD[1:0] set to 1_H or 2_H, observe the following notes.

- For the start virtual channel of each scan group, make a setting so as not to use an external multiplexer. (However, for the start virtual channel of the scan group whose priority is lowest among the scan groups for which the external multiplexer is used, a setting to use an external multiplexer does not cause any problem.)
- Up to two scan groups can be transferred with the SPI interface.

Furthermore, set the MPX wait as follows:

- When an MPX value is transferred at a port: Insert a wait of at least 1 µs.
- When an MPX value is transferred with the SPI interface: Insert a wait of at least SPI transmission time × 2.

26.9.2 Notes on Using Analog Input Pins

- Do not perform A/D conversion for an analog pin with both ΔΣADC and SAR-ADC at the same time. Also, do not perform A/D conversion with SAR-ADC for an analog pin that is being selected for digital input. Doing so may degrade the A/D conversion accuracy.

- When a digital input or output signal is multiplexed with an analog input signal, it can also be used as a digital general-purpose input or output pin. Changes in a digital input or output during AD conversion may reduce the precision of conversion. Noise from the operation of digital pins near analog input pins may also reduce the precision of conversion. Notes on how to reduce the effects of digital input and output on the results of AD conversion are given below.

(1) Notes on Analog Input Pins

(a) Place the capacitors of RC circuits as close to the LSI pin as is possible. This reduces the effects of digital inputs and outputs on the precision of conversion. Since the improvement in precision also depends on other conditions of the board, evaluate the situation on the actual board.

(2) Notes on Digital Input and Output Pins near Analog Pins

(a) If digital input through the pin is not essential, disable digital input.

(b) If a digital signal is input to such a pin, the signal should not include overshot or undershot.

(c) Set the voltage to AVCC = EVCC (if an analog pin function is multiplexed with the digital function).
Operation outside the range of -0.3 V AVCC-EVCC 0.3 V may negatively affect the reliability of the chip.

(d) Design the board so that load capacitances connected to output pins to suppress discharge currents are small.

(e) Lower the output driving ability (boosting) of pins that may be affected.

(3) Software Measures against Effects on the Results of Conversion

(a) Use the average of multiple results of AD conversion.

(b) When using multiple consecutive results of ADC conversion, exclude outlying results.

26.9.3 Notes when Current is Being Injected

Proceeding with A/D conversion by using simultaneous track-and-hold while current is being injected may affect the accuracy of A/D conversion.

To avoid the effects of injection current on the accuracy of A/D conversion, make settings according to either of the following descriptions.

CAUTION

Applying a voltage above the power supply voltage or below the ground voltage to a pin will lead to an injection current flowing.

(1) If you are using simultaneous track-and-hold, do not allow current to be injected to pins where this will affect the accuracy of conversion.

Table 26.42 Combination of the T&H Circuits and Pins where Injected Current will Affect the Accuracy of Conversion

Unit	T&H Circuit for Conversion	Pins where Injected Current will Affect the Accuracy of Conversion
ADCB0	T&H Circuit 0	AN000
	T&H Circuit 1	AN001
ADCB1	T&H Circuit 0	AN100
	T&H Circuit 1	AN101

(2) When simultaneous track-and-hold is in use, setting a value other than 00_B to bits 4 and 3 (GCTRL) of the ADCBnVCRj register changes the pins where injected current will affect the accuracy of conversion.
The table below shows the correspondence between settings other than 00_B for bits 4 and 3 (GCTRL) of the ADCBnVCRj register and the pins which are subject to injected current affecting the accuracy of conversion. Follow the table to set bits 4 and 3 (GCTRL) such that the result for pins which may be affected is no applicable pins or use pins to which no injection current is being applied.

Table 26.43 Combination of the T&H Circuits and Pins where Injected Current will Affect the Accuracy of Conversion (the ADCBnVCRj.GCTRL bits are used)

Unit	T&H Circuit for Conversion	Settings Other than 00_B for GCTRL[4:3]	Pins where Injected Current will Affect the Accuracy of Conversion
ADCB0	T&H Circuit 0	11_B	– (no applicable pins)
	T&H Circuit 1	11_B	– (no applicable pins)
ADCB1	T&H Circuit 0	00_B	AN100
		01_B	AN120
		10_B	AN140
		11_B	AN160
	T&H Circuit 1	00_B	AN101
		01_B	AN121
		10_B	AN141
		11_B	AN161

26.10 IFC (Integer/Floating-Point Conversion Module)

The IFC module can read data in the ADC's register data via the IFC and convert it to the floating-point format.

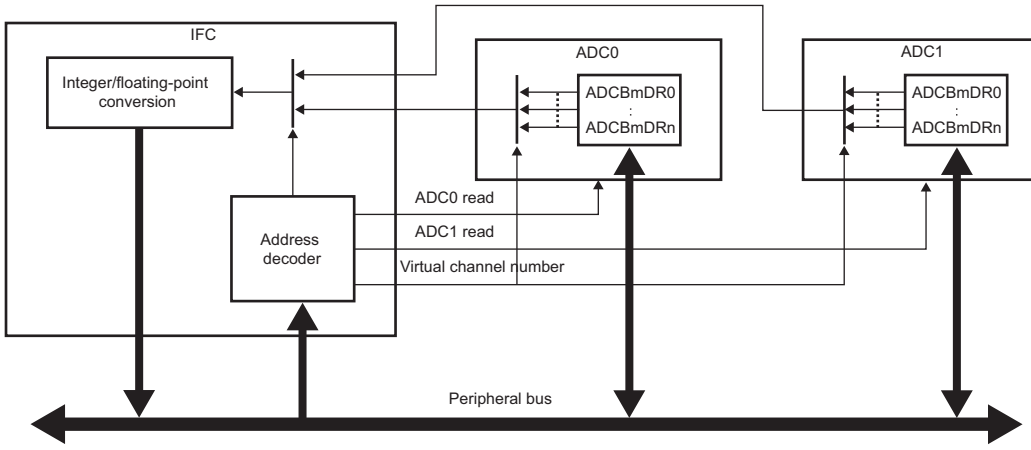
26.10.1 Features

The IFC has the following features:

- Floating-point format: IEEE754 standard single-precision
- Supported registers: Data registers for all virtual channels of the ADC

26.10.2 Configuration

Figure 26.27 illustrates the IFC block diagram.



Note: n = 39

Figure 26.27 IFC Block Diagram

26.10.3 Register Address

Table 26.44 Register Base Address <IFC_base>

<IFC_base> Address
FFF2 3000_H

Table 26.45 Register Address List

Register Name	Symbol	Address
Floating-point data register mn	FDRmn	<IFC_base> + m × 100_H + n × 4_H

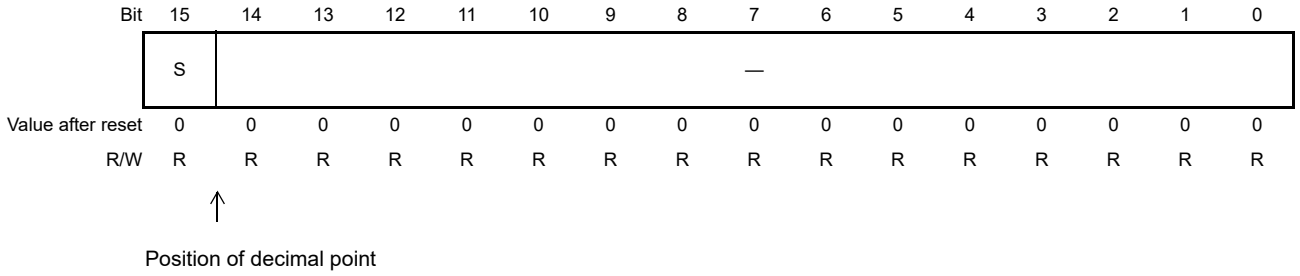
Note: m = 0, 1; n = 0 to 39

26.10.4 Register

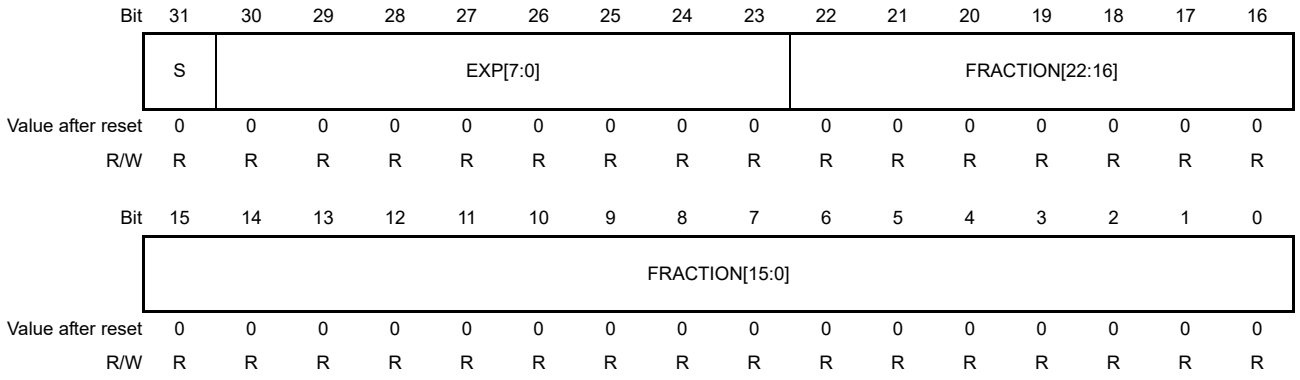
26.10.4.1 FDRmn — Floating-Point Data Register mn

FDRmn is a 32-bit read-only register that can read DRn of ADCm, converting it to the floating-point format. Data in an ADC data register is converted from the signed fixed-point format to the floating-point format. Even data in the signed integer format (DFMT bit in ADCBmADCR2 of ADCm is set to 1) is regarded as data in the signed fixed-point format, and converted to the floating-point format.

Signed fixed-point format



Floating-point format



Bit Position	Bit Name	Function
31	S	Sign bit
30 to 23	EXP[7:0]	Exponent
22 to 0	FRACTION[22:0]	Mantissa

Expression: (-1)^S × 2^ (EXP-127) × (1+ (Fraction × 2^ (-23)))

NOTE

m = 0, 1; n = 0 to 39

26.11 ADC Summation Function (ASF)

The ASF is a function that accumulates the A/D converted value from the ADC the specified number of times, and then stores the total value in a register.

26.11.1 Features

The ASF has the following features:

- Compatible module: ADC0
- Number of accumulation channels: 11
- Entry from ADC:
  Upon receiving an ASF entry request, TAG for DFE, and data for DFE from the ADC, the ASF enters them to each accumulation channel (with the same number as the TAG value for DFE) of the ASF.
- Accumulation data register: 32 bits
- Accumulation end interrupt:
  Each time accumulation for the number of times specified in each channel ends, an interrupt request (ASI0 to ASI10) to the CPU can be generated.

26.11.2 Configuration

Figure 26.28 illustrates the ASF block diagram.

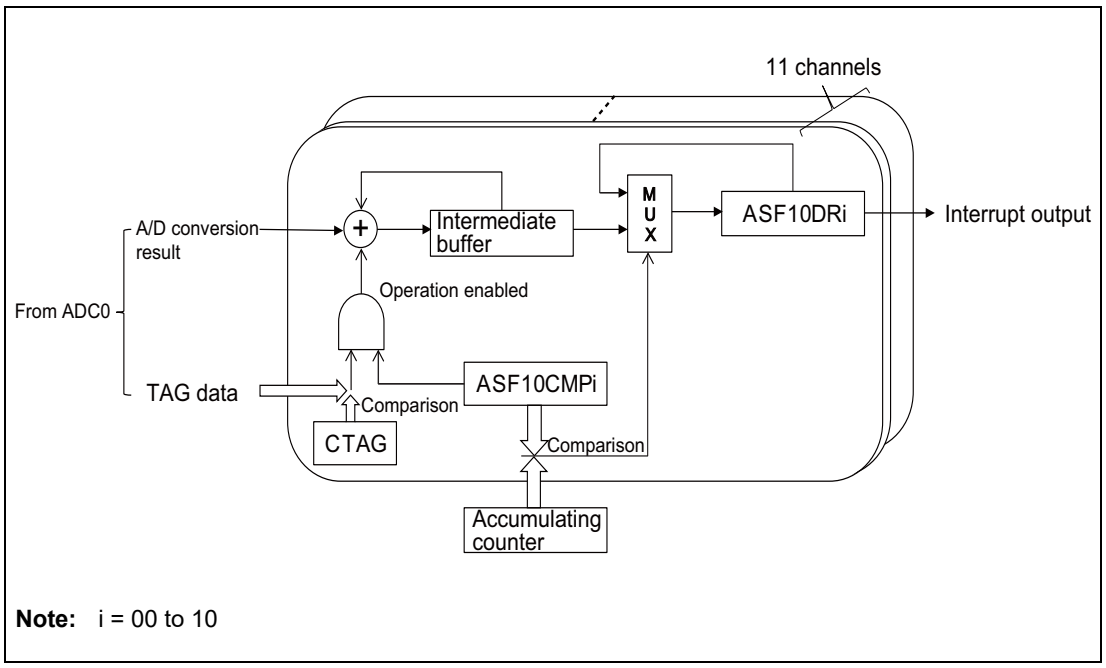


Figure 26.28 Block Diagram of ASF Configuration

26.11.3 Register Addresses

Table 26.46 Register Base Address <ASF_base>

<ASF_base> Address
FFF2 2000_H

Table 26.47 List of Register Addresses

Register Name	Symbol	Value after Reset	Address	Access Size
Accumulation Data Read Register 0	ASF1nDR00	0000 0000_H	<ASF_base> + 00_H	32
Accumulation Data Read Register 1	ASF1nDR01	0000 0000_H	<ASF_base> + 04_H	32
Accumulation Data Read Register 2	ASF1nDR02	0000 0000_H	<ASF_base> + 08_H	32
Accumulation Data Read Register 3	ASF1nDR03	0000 0000_H	<ASF_base> + $0C_H$	32
Accumulation Data Read Register 4	ASF1nDR04	0000 0000_H	<ASF_base> + 10_H	32
Accumulation Data Read Register 5	ASF1nDR05	0000 0000_H	<ASF_base> + 14_H	32
Accumulation Data Read Register 6	ASF1nDR06	0000 0000_H	<ASF_base> + 18_H	32
Accumulation Data Read Register 7	ASF1nDR07	0000 0000_H	<ASF_base> + $1C_H$	32
Accumulation Data Read Register 8	ASF1nDR08	0000 0000_H	<ASF_base> + 20_H	32
Accumulation Data Read Register 9	ASF1nDR09	0000 0000_H	<ASF_base> + 24_H	32
Accumulation Data Read Register 10	ASF1nDR10	0000 0000_H	<ASF_base> + 28_H	32
Accumulation Compare Match Register 0	ASF1nCMP00	00_H	<ASF_base> + 40_H	8
Accumulation Compare Match Register 1	ASF1nCMP01	00_H	<ASF_base> + 44_H	8
Accumulation Compare Match Register 2	ASF1nCMP02	00_H	<ASF_base> + 48_H	8
Accumulation Compare Match Register 3	ASF1nCMP03	00_H	<ASF_base> + $4C_H$	8
Accumulation Compare Match Register 4	ASF1nCMP04	00_H	<ASF_base> + 50_H	8
Accumulation Compare Match Register 5	ASF1nCMP05	00_H	<ASF_base> + 54_H	8
Accumulation Compare Match Register 6	ASF1nCMP06	00_H	<ASF_base> + 58_H	8
Accumulation Compare Match Register 7	ASF1nCMP07	00_H	<ASF_base> + $5C_H$	8
Accumulation Compare Match Register 8	ASF1nCMP08	00_H	<ASF_base> + 60_H	8
Accumulation Compare Match Register 9	ASF1nCMP09	00_H	<ASF_base> + 64_H	8
Accumulation Compare Match Register 10	ASF1nCMP10	00_H	<ASF_base> + 68_H	8
Accumulation Counter Control Register 0	ASF1nCTL0	0000_H	<ASF_base> + 80_H	8, 16
Accumulation Counter Control Register 1	ASF1nCTL1	00_H	<ASF_base> + 84_H	8
Accumulation Count Read Register	ASF1nCNT	00_H	<ASF_base> + $8C_H$	8

Note: n = 0

26.11.4 Registers

26.11.4.1 ASF1nDRi — Accumulation Data Read Register i

ASF1nDRi is a 32-bit read-only register that stores an accumulation value. When ASF1nCNT matches ASF1nCMPi, data in accumulator i is updated to ASF1nDRi. The ASF1nDRi value is retained until the next compare match occurs.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ASF1nDRi[16:1]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ASF1nDRi[0:0]	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

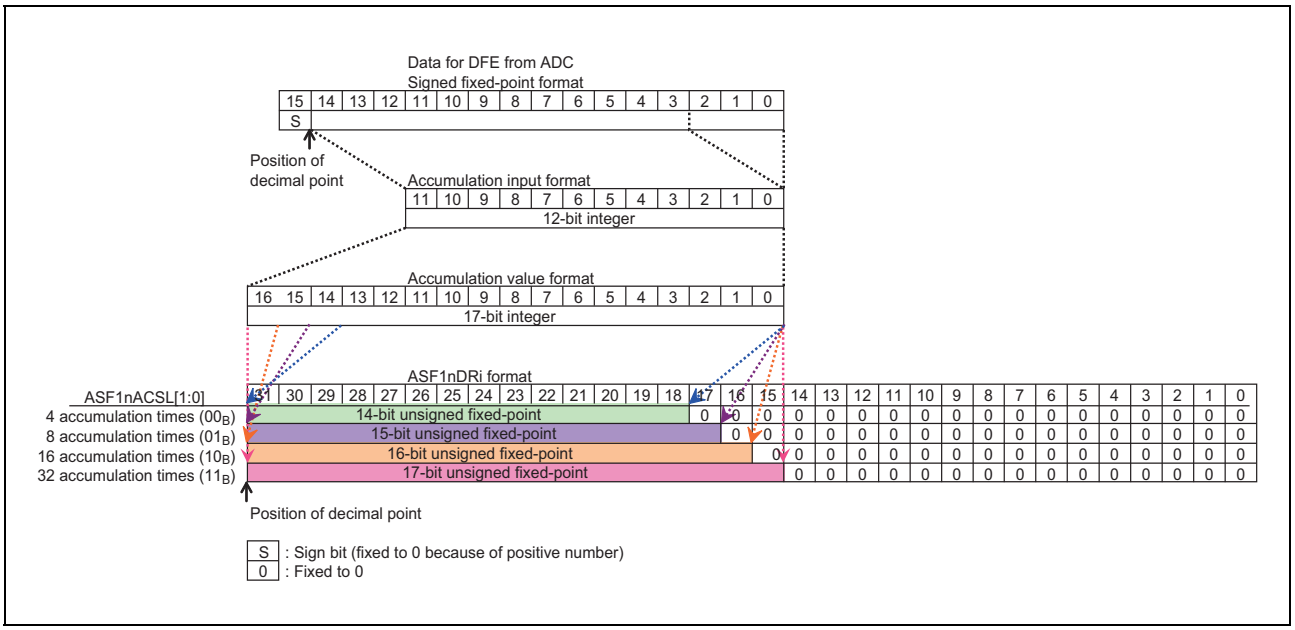
Table 26.48 ASF1nDRi Register Contents

Bit Position	Bit Name	Function
31 to 15	ASF1nDRi [16:0]	Accumulation Data When ASF1nCNT matches ASF1nCMPi, data in accumulator i is updated to ASF1nDRi.

Note: i = 00 to 10 or 0 to 10
n = 0

A 12-bit (bit14 to bit3) value is extracted from the data for DFE from the ADC for use in the input of accumulation as a 12-bit integer. Therefore, even if the valid number of digits of the data for DFE is 13 or 14 bits after addition is performed twice or four times in the ADC, the valid number of digits is regarded as 12 bits and accumulation processing is performed. Even the signed integer format (DFMT = 1 is set in the ADC) is regarded as the signed fixed-point format and accumulation processing is performed. For this reason, when using the ASF function, be sure to disable the ADC's addition function and set DFMT to 0 (signed fixed-point format).

ASF1nDRi is in a format with left-aligned valid bits according to the accumulation count setting out of 17-bit accumulation value. ASF1nDRi is initialized to 0000 0000_H at a reset.



26.11.4.2 ASF1nCMPi — Accumulation Compare Match Register i

ASF1nCMPi is an 8-bit readable/writable register that enables or disables accumulation processing for accumulation channel i and specifies the ASF1nCNT compare match value for updating ASF1nDRi. ASF1nCMPi is initialized to 00_H at a reset.

Bit	7	6	5	4	3	2	1	0
	ASF1nCHEi	—	—	ASF1nCMPi[4:0]				
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W

Table 26.49 ASF1nCMPi Register Contents

Bit Position	Bit Name	Function
7	ASF1nCHEi	Accumulation Channel i Enable 0: Accumulation is disabled. – Accumulator i is cleared to 0. – ASF1nDRi is not updated. – Accumulation end interrupt i (INT_ACE[i]) is not generated. 1: Accumulation is enabled. – Each time an ASF entry request of TAG for DFE (TAG for DFE = i) that matches accumulation channel i is accepted, the result of sum of data for DFE and accumulator i is stored in accumulator i (accumulation processing for accumulation channel i). – When ASF1nCNT matches ASF1nCMPi, data in accumulator i is updated to ASF1nDRi and accumulator i is cleared to 0 (ASF1nDRi update). – An accumulation end interrupt i (INT_ACE[i]) is generated on compare match.
4 to 0	ASF1nCMPi[4:0]	Accumulation Compare Match When ASF1nCNT matches ASF1nCMPi, data in accumulator i is updated to ASF1nDRi. These bits must not be modified during operation (while ASF1nST = 1).

Note: i = 00 to 10 or 0 to 10
n = 0

CAUTIONS

1. Each time ASF1nCNT[4:0] matches ASF1nCMPi[4:0], the accumulator i value is stored in ASF1nDRi[16:0] and an INT_ACE[i] interrupt is generated.
   Therefore, the first accumulation count in ASF1nDRi[16:0] is not ensured after accumulation processing (where ASF1nST is changed from 0 to 1) is started, or immediately after an accumulation channel (for which ASF1nCHEi is changed from 0 to 1 during accumulation processing) is enabled (the accumulation count is variable).
   For example of operation, see **Section 26.11.5.1, Example of Accumulation Processing Operation**.
2. Set ASF1nCMPi[4:0] to a value less than the accumulation count selected by ASF1nACSL[1:0]. If a value equal to or greater than the accumulation count is set in ASF1nCMPi[4:0], ASF1nCNT[4:0] does not match ASF1nCMPi[4:0], so the ASF1nDRi[16:0] value is not updated.

26.11.4.3 ASF1nCTL0 — Accumulation Counter Control Register 0

ASF1nCTL0 is a 16-bit or 8-bit readable/writable register that sets count-up conditions for ASF1nCNT. ASF1nCTL0 is initialized to 0000_H at a reset.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	ASF1nACSL [1:0]		—	—	—	—	ASF1nCTAG[3:0]			
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 26.50 ASF1nCTL0 Register Contents

Bit Position	Bit Name	Function
9, 8	ASF1nACSL[1:0]	Accumulation Count Select 00_B: 4 times 01_B: 8 times 10_B: 16 times 11_B: 32 times These bits must not be modified during operation (while ASF1nST = 1).
3 to 0	ASF1nCTAG[3:0]	Accumulation Count-Up TAG When an ASF entry request of TAG for DFE that matches ASF1nCTAG is accepted, ASF1nCNT starts count-up. These bits must not be modified during operation (while ASF1nST = 1).

Note: n = 0

26.11.4.4 ASF1nCTL1 — Accumulation Counter Control Register 1

ASF1nCTL1 is an 8-bit readable/writable register that enables (starts) or disables (stops) counting up by ASF1nCNT. ASF1nCTL1 is initialized to 00_H at a reset.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ASF1nST
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 26.51 ASF1nCTL1 Register Contents

Bit Position	Bit Name	Function
0	ASF1nST	Accumulation Counter Start 0: The accumulation counter and all accumulators are cleared to 0. 1: Counting up and accumulation processing by the accumulation counter are enabled. When an ASF entry request of TAG for DFE that matches ASF1nCTAG is accepted with ASF1nST set to 1, ASF1nCNT starts counting up.

Note: n = 0

26.11.4.5 ASF1nCNT — Accumulation Count Read Register

ASF1nCNT is an 8-bit read-only register that indicates the count value of the accumulation counter. ASF1nCNT is initialized to 00_H at a reset.

Bit	7	6	5	4	3	2	1	0
	—	—	—	ASF1nCNT[4:0]				
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 26.52 ASF1nCNT Register Contents

Bit Position	Bit Name	Function
4 to 0	ASF1nCNT[4:0]	Accumulation Counter When ASF1nST is set to 1, ASF1nCNT counts up each time an ASF entry request of TAG for DFE that matches ASF1nCTAG is accepted. When ASF1nST is set to 0, ASF1nCNT is cleared to 0. • When ASF1nACSL = 00_B (4 accumulation times) The counter counts up to 3 and then returns to 0. • When ASF1nACSL = 01_B (8 accumulation times) The counter counts up to 7 and then returns to 0. • When ASF1nACSL = 10_B (16 accumulation times) The counter counts up to 15 and then returns to 0. • When ASF1nACSL = 11_B (32 accumulation times) The counter counts up to 31 and then returns to 0.

Note: n = 0

26.11.5 Operation

The summation function ASF accumulates the results of ADC conversion from the ADC, stores the accumulation value in a register, and generates an interrupt request each time accumulation for the specified number of times is completed.

The ASF allows accumulation for up to 11 channels. The number of accumulation times is selectable from 4, 8, 16, and 32. However, the number of accumulation times is common to all channels (it cannot be set individually for each channel). Furthermore, interrupt timing offset is enabled by register setting.

After ASF1nST is set to 1 to enable accumulation processing, the ADC conversion result is added to the accumulation channel that is the same as the TAG value for DFE each time an ASF entry request from the ADC becomes a high level.

The accumulation counter ASF1nCNT[4:0] counts up when TAG for DFE that matches ASF1nCTAG[3:0] is input and an ASF entry request becomes a high level. When ASF1nCTAG [3:0] matches the TAG for DFE and the accumulation count reaches number of accumulation times specified by ASF1nACSL[1:0] - 1, the accumulation counter returns to 1 at an ASF entry request timing.

When ASF1nCMPi[4:0] matches ASF1nCNT[4:0], the accumulation data ASF1nDRi[31:0] is updated and an accumulation end interrupt ASI[i] is generated.

[Setting Procedure]

1. While ASF1nST = 0, set a required value in each of ASF1nACSL[1:0], ASF1nCTAG[3:0], ASF1nCMPi [4:0], and ASF1nCHEi in any order.
2. Set ASF1nST to 1 to enable (start) the accumulation processing operation.
3. Ignore the first accumulation end interrupt ASI[i] (or discard the ASF1nDRi[16:0] read value) because the first accumulation count after accumulation start is variable.
4. Read ASF1nDRi[16:0] required for accumulation end interrupt ASI[i] from the second time, and then read the accumulation value of the ADC conversion result.
5. After the required accumulation value of the ADC conversion results is obtained, set ASF1nST to 0 to finish the accumulation processing.

26.11.5.1 Example of Accumulation Processing Operation

(1) Operation start

Figure 26.29 shows an example of accumulation processing operation for four channels when the number of accumulation times is set to 8 (ASF1nACSL[1:0] = 01_B), ASF1nCTAG[3:0] = 1_H, ASF1nCMP00[4:0] = 00_H, ASF1nCMP01[4:0] = 01_H, ASF1nCMP02[4:0] = 02_H, and ASF1nCMP03[4:0] = 03_H.

In this example, data of TAG (for DFE) = 0_H is always 10, data of TAG (for DFE) = 1_H is always 20, data of TAG (for DFE) = 2_H is always 30, and data of TAG (for DFE) = 3_H is always 40 for convenience.

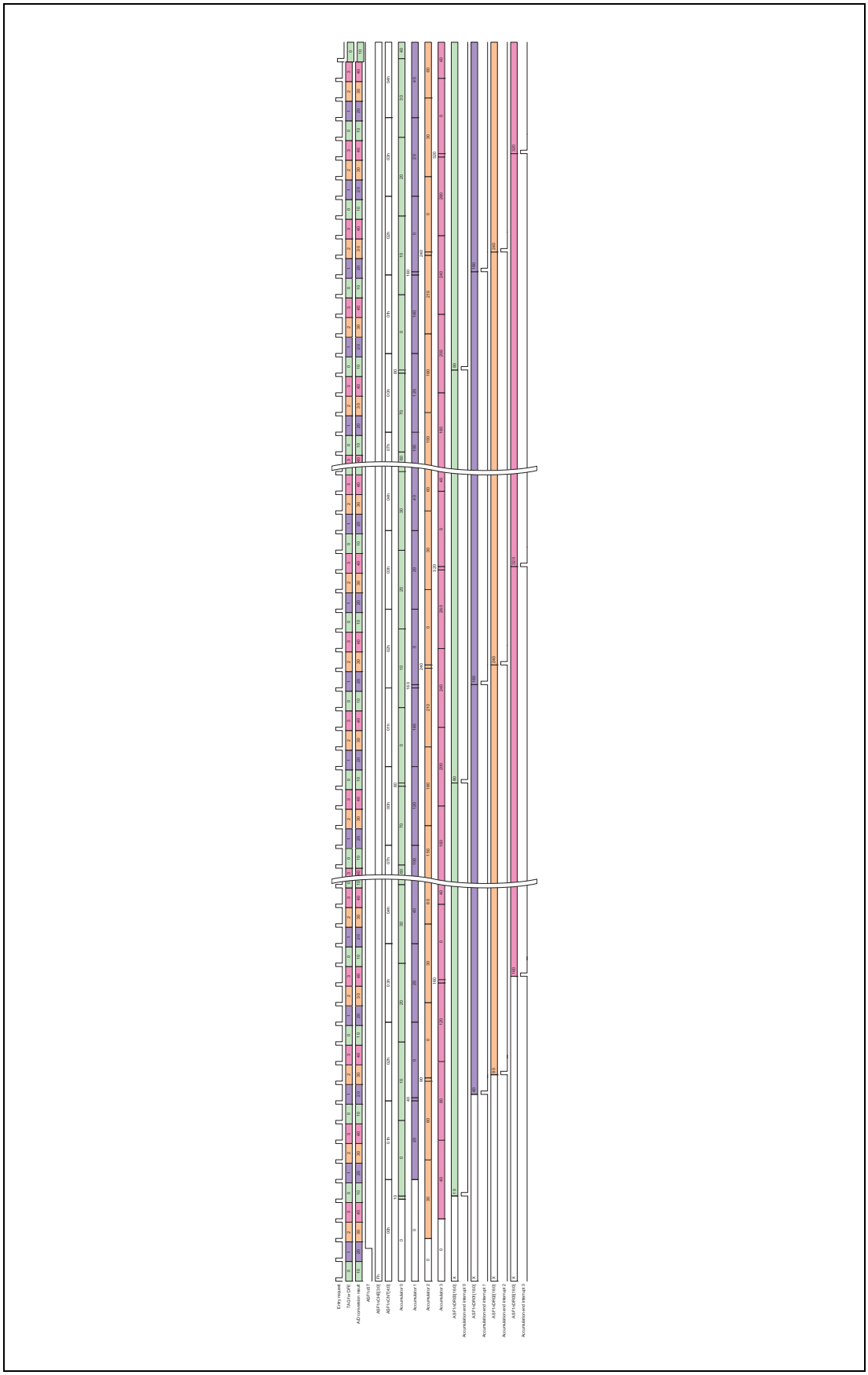


Figure 26.29 Timing of Accumulation Processing Start Operation Example

(2) Operation stop and restart

Figure 26.30 shows an example of accumulation processing stop and restart operation.

When ASF1nST is set to 0, the accumulation counter ASF1nCNT[4:0] and accumulator i are cleared to 0.

Figure 26.30 Timing of Accumulation Processing Stop and Restart Operation Example

(3) Channel stop

Figure 26.31 shows an example of accumulation processing stop and restart operation for an accumulation channel.

When ASF1nCHEi is set to 0, accumulator i is cleared to 0.

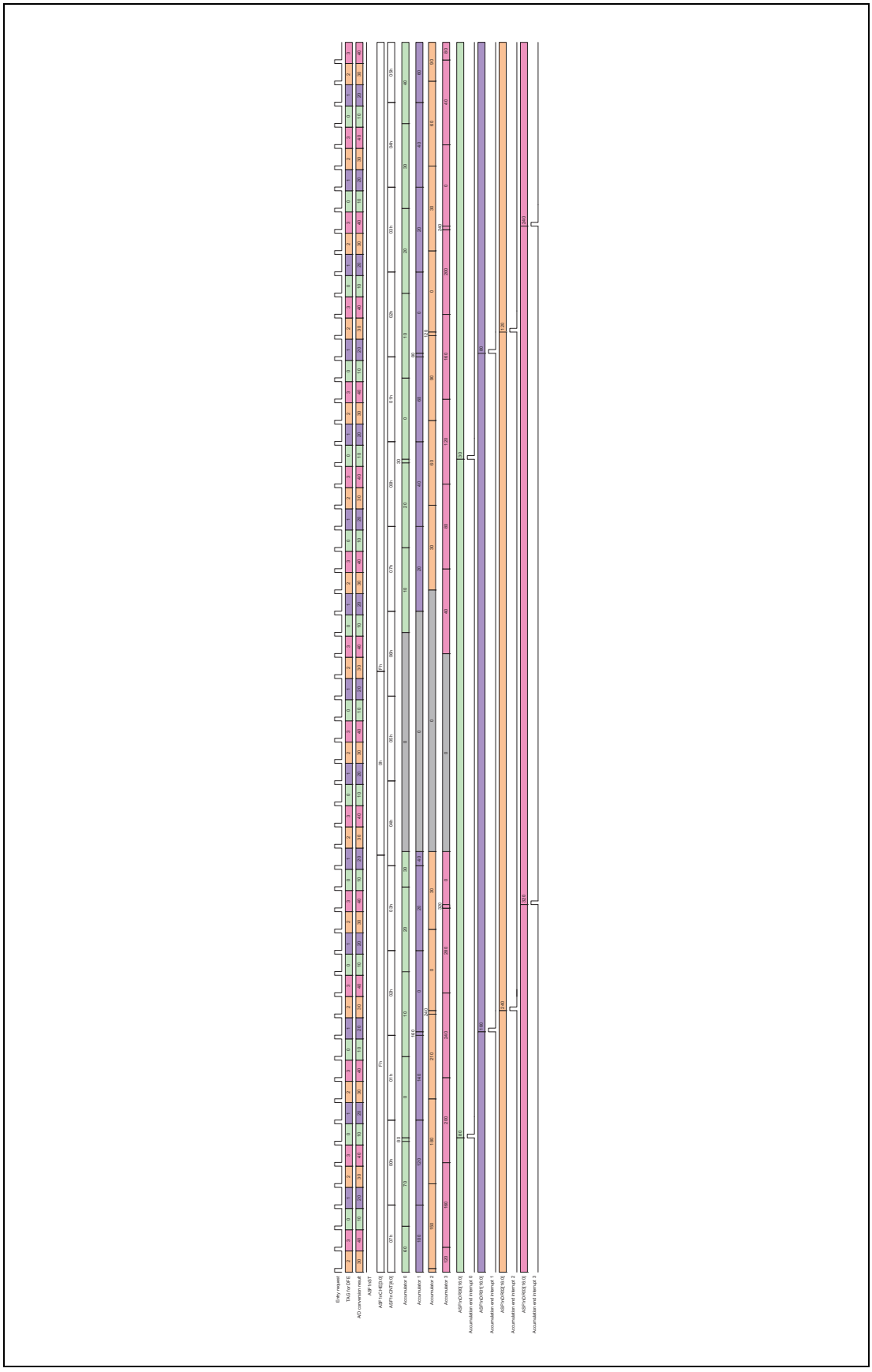


Figure 26.31 Timing of Accumulation Processing Stop Operation Example by ASF1nCHEi

Section 27 Delta-Sigma AD Converter (DS-ADC)

This LSI, provided with eight A/D converter modules (DSADC0 to DSADC7) using the delta-sigma modulation method, consists of a delta-sigma modulator and a digital filter.

27.1 Features

The following describes features of the delta-sigma ADC.

- Analog channels
  Eight channels are provided.
- Enhanced A/D converter
  - Valid bits (analog signal bandwidth):
    13 bits (30 kHz) when Fos = 8 Msps, Fs = 100 ksps (usage example 1)
    13 bits (30 kHz) when Fos = 8 Msps, Fs = 200 ksps (usage example 2)
    13 bits (60 kHz) when Fos = 8 Msps, Fs = 200 ksps (usage example 3)
    8 bits (200 kHz) when Fos = 8 Msps, Fs = 1.6 Msps (usage example 4)
  - A/D conversion method: Delta-sigma modulation method
  - Oversampling rate (Fos): 8 Msps
  - Sampling rate (Fs): 100 ksps, 200 ksps, 1.6 Msps
- Supporting single-ended input and differential input
  Single-ended input or differential input can be selected for each delta-sigma ADC (DSADC0 to DSADC7).
- Input gain function
  Each delta-sigma ADC (DSADC0 to DSADC7) contains a programmable gain amplifier (PGA). A gain of ×1, ×2, ×4, or ×8 is selectable.
- Data register
  A data register corresponding to each delta-sigma ADC (DSADC0 to DSADC7) is contained in the data supplementary information register.
- A/D conversion start trigger
  Each delta-sigma ADC (DSADC0 to DSADC7) can start A/D conversion by software, each timer trigger, or an external trigger (DSADC0 to DSADC7: DSADTRG0 to DSADTRG7).
- A/D conversion end trigger
  Each delta-sigma ADC can terminate A/D conversion by software, each timer trigger, or an external trigger ((DSADC0 to DSADC7: DSADTRG0 to DSADTRG7).
- Entry to the digital filter engine
  A/D converted values can be entered directly in the digital filter engine (DFE).
  Whether to enable or disable DFE entry and TAG to determine channels for entry can be set for each delta-sigma ADC (DSADC0 to DSADC7).
- Supporting DMA transfer based on A/D conversion end timing
  Each time A/D conversion ends, each delta-sigma ADC (DSADC0 to DSADC7) outputs a DMA request on completion of A/D conversion (DSADIm), which can generate DMA request sources (DSADC0 to DSADC7: DSADI0 to DSADI7) to the DMAC or activate the DMAC.
- AD error interrupt request and AD parity error notification
  Each delta-sigma ADC (DSADC0 to DSADC7) can generate an AD error interrupt request

(DSADEm) to the INTC and an AD parity error notification (DSADPEm) to the ECM (error control module).

- Settable analog conversion voltage range
  An analog conversion voltage range can be set using the ADSVREFH pin and the ADSVREFL pin. When performing A/D conversion by using the single-ended input, either ADSVREFL or ADSVREFH/2 is selectable as a common voltage.
- A/D conversion monitor output
  A/D conversion timing can be output to the DSADEND0 to DSADEND7 pins.
- A wide range of safety functions
  A variety of safety functions are provided, such as self-diagnosis of delta-sigma ADC, pin-level self-diagnosis, upper-limit/lower-limit check for data registers, data register parity check for data registers, overwrite check for data registers, and read and clear function for data registers.

27.2 Configuration

Figure 27.1 is a block diagram of the delta-sigma ADC, and **Table 27.1** shows the pin configuration of the delta-sigma ADC.

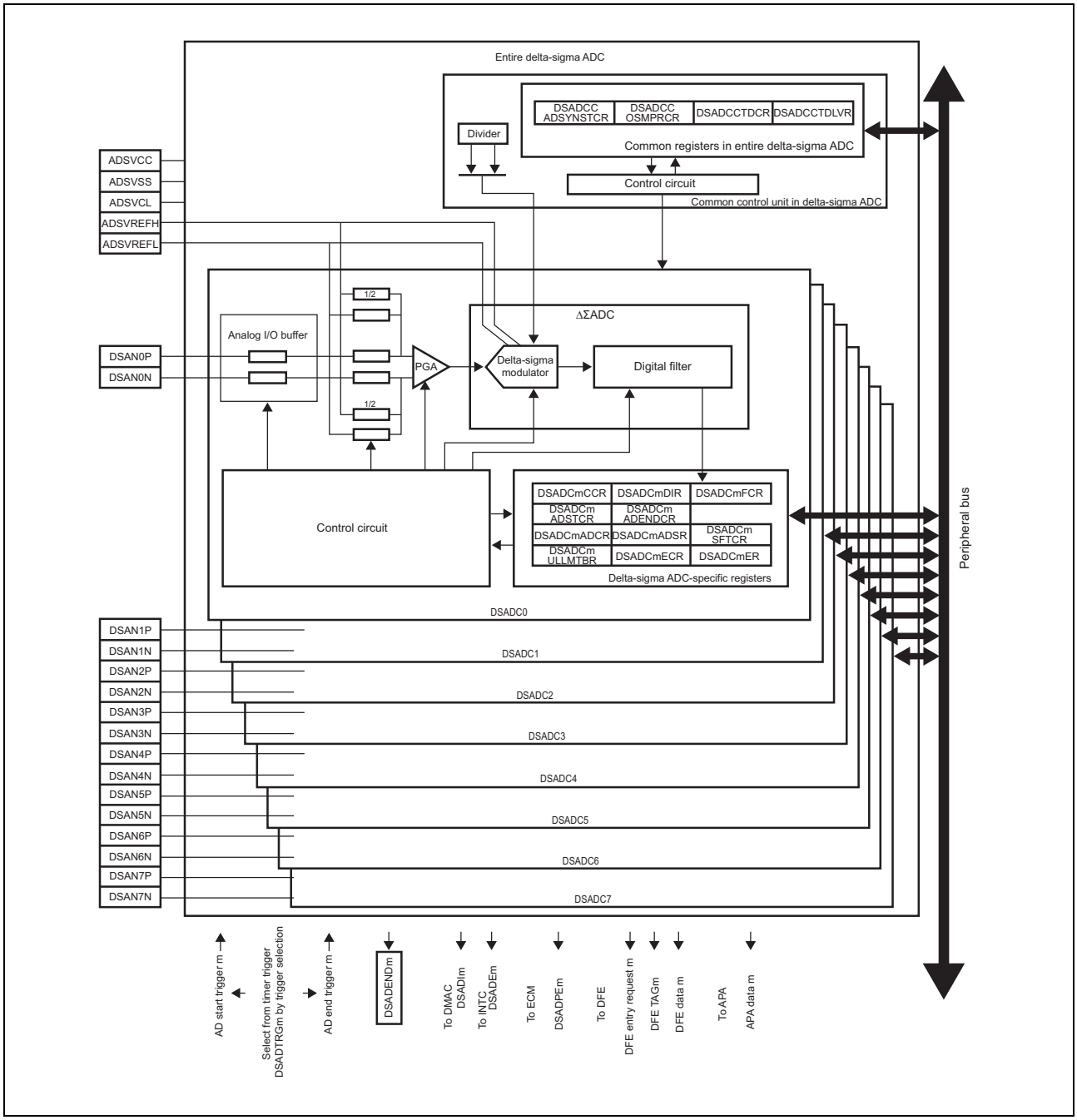


Figure 27.1 Block Diagram of Delta-Sigma ADC

Table 27.1 Delta-Sigma ADC Pin Configuration

Pin Name	I/O	Description
ADSVCC	Input	Analog block power supply pin
ADSVSS	Input	Analog block ground pin
ADSVREFH	Input	Analog block upper reference voltage pin
ADSVREFL	Input	Analog block lower reference voltage pin
ADSVCL	—	Delta-sigma ADC external capacitor pin
DSAN0P	Input	Analog pin 0P
DSAN0N	Input	Analog pin 0N
DSAN1P	Input	Analog pin 1P
DSAN1N	Input	Analog pin 1N
DSAN2P	Input	Analog pin 2P
DSAN2N	Input	Analog pin 2N
DSAN3P	Input	Analog pin 3P
DSAN3N	Input	Analog pin 3N
DSAN4P	Input	Analog pin 4P
DSAN4N	Input	Analog pin 4N
DSAN5P	Input	Analog pin 5P
DSAN5N	Input	Analog pin 5N
DSAN6P	Input	Analog pin 6P
DSAN6N	Input	Analog pin 6N
DSAN7P	Input	Analog pin 7P
DSAN7N	Input	Analog pin 7N
DSADTRG0	Input	External trigger pin (for DSADC0)
DSADTRG1	Input	External trigger pin (for DSADC1)
DSADTRG2	Input	External trigger pin (for DSADC2)
DSADTRG3	Input	External trigger pin (for DSADC3)
DSADTRG4	Input	External trigger pin (for DSADC4)
DSADTRG5	Input	External trigger pin (for DSADC5)
DSADTRG6	Input	External trigger pin (for DSADC6)
DSADTRG7	Input	External trigger pin (for DSADC7)
DSADEND0	Output	A/D conversion timing monitor pin (for DSADC0)
DSADEND1	Output	A/D conversion timing monitor pin (for DSADC1)
DSADEND2	Output	A/D conversion timing monitor pin (for DSADC2)
DSADEND3	Output	A/D conversion timing monitor pin (for DSADC3)
DSADEND4	Output	A/D conversion timing monitor pin (for DSADC4)
DSADEND5	Output	A/D conversion timing monitor pin (for DSADC5)
DSADEND6	Output	A/D conversion timing monitor pin (for DSADC6)
DSADEND7	Output	A/D conversion timing monitor pin (for DSADC7)

To ensure reliability of this LSI, the following relationship is required between ADSVCC and ADSVSS and VCC and VSS when using a delta-sigma ADC.

ADSVCC = 5 V ± 0.5 V, ADSVSS = VSS

When no delta-sigma ADC is used, do not leave the ADSVCC and ADSVSS pins open. The voltage to be applied to analog input pins must be within the following range.

ADSVREFL ≤ DSANkx (k = 0 to 7 x = P, N) ≤ ADSVREFH

27.3 Register Addresses

Table 27.2 Register Base Address (<DSADCcom_base>, <DSADCm_base>)

DSADCcom DSADCm	<DSADCcom_base> address <DSADCm_base> address
DSADCcom	FFF3 0000_H
DSADC0	FFF3 1000_H
DSADC1	FFF3 2000_H
DSADC2	FFF3 3000_H
DSADC3	FFF3 4000_H
DSADC4	FFF3 5000_H
DSADC5	FFF3 6000_H
DSADC6	FFF3 7000_H
DSADC7	FFF3 8000_H

Table 27.3 Register Addresses

Register Name	Symbol	Value after Reset	Address	Access Size
Delta-sigma ADC common registers				
AD synchronization start control register	DSADCCADSYNSTCR	00_H	<DSADCcom_base> + 00_H	8
Oversampling rate control register	DSADCCOSMPRCR	00_H	<DSADCcom_base> + 04_H	8
Pin level self-diagnosis control register	DSADCCTDCR	00_H	<DSADCcom_base> + 08_H	8
Pin level self-diagnosis level setting register	DSADCCTDLVR	0000_H	<DSADCcom_base> + $0C_H$	8, 16
Delta-sigma ADC-specific registers (m = 0 to 7)				
Channel control register	DSADCmCCR	0000 0000_H	<DSADCm_base> + 00_H	8, 16, 32
Data supplementary information register	DSADCmDIR	0000 0000_H	<DSADCm_base> + 04_H	32
AD start control register	DSADCmADSTCR	00_H	<DSADCm_base> + 08_H	8
AD stop control register	DSADCmADENDCR	00_H	<DSADCm_base> + $0C_H$	8
AD control register	DSADCmADCR	00_H	<DSADCm_base> + 10_H	8
AD status register	DSADCmADSR	00_H	<DSADCm_base> + 14_H	8
Safety control register	DSADCmSFTCR	00_H	<DSADCm_base> + 18_H	8
Upper-limit/lower-limit table register	DSADCmULLMTBR	7FFF 8000_H	<DSADCm_base> + $1C_H$	16, 32
Error clear register	DSADCmECR	00_H	<DSADCm_base> + 20_H	8
Error register	DSADCmER	00_H	<DSADCm_base> + 24_H	8
Digital filter FIR control register	DSADCmFCR	00_H	<DSADCm_base> + 28_H	8

NOTE

DSADCcom_base: Common register base address

DSADCm_base: Specific register base address (DSADCm register)

m = 0 to 7

27.4 Common Registers for Delta-Sigma ADC

The following describes registers shared by all delta-sigma ADCs (DSADC0 to DSADC7).

27.4.1 DSADCCADSYNSTCR — AD Synchronization Start Control Register

DSADCCADSYNSTCR is an 8-bit write-only register to control simultaneous start of delta-sigma ADCs. This register is always read as 0.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ADSTART
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 27.4 DSADCCADSYNSTCR Register Contents

Bit Position	Bit Name	Function
7 to 1	—	Reserved These bits are always read as 0. The write value should always be 0.
0	ADSTART	A/D Conversion Start of Delta-Sigma ADCs Condition for starting A/D conversion of DSADCm by ADSTART A value of 1 is written to ADSTART when ADTACT for DSADCm is 0 and ADSTARTE is 1. Delta-sigma ADCs for which ADSTARTE is set to 1 are activated simultaneously. While the setting of ADACT in DSADCm is 1, writing 1 to this bit is ignored and DSADCm continues A/D conversion in progress. While the setting of ADSTARTE in DSADCm is 0, writing 1 to this bit is ignored and DSADCm does not start A/D conversion. Writing 0 to this bit is ignored.

Note 1. m = 0 to 7

Note 2. It is recommended that ADSTART be written to while ENDTRGE for all the delta-sigma ADCs is 0 considering external trigger.

27.4.2 DSADCCOSMPRCR — Oversampling Rate Control Register

DSADCCOSMPRCR is an 8-bit readable/writable register to control the oversampling rate.
DSADCCOSMPRCR is initialized to 00_H by a reset.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	OSMPR
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 27.5 DSADCCOSMPRCR Register Contents

Bit Position	Bit Name	Function
7 to 1	—	Reserved These bits are always read as 0. The write value should always be 0.
0	OSMPR	Oversampling Rate This bit selects the oversampling rate. When a delta-signal ADC is to be used, be sure to set 1_B = 8 MHz. 0: Setting prohibited 1: 8 Msps

CAUTION

To prevent malfunction, DSADCCOSMPRCR should be set while ADACT for all delta-sigma ADCs is 0 (before starting A/D conversion) and STTRGE for all delta-sigma ADCs is 0.

27.4.3 DSADCCTDCR — Pin Level Self-Diagnosis Control Register

DSADCCTDCR is an 8-bit readable/writable register to control self-diagnosis of pin levels.
DSADCCTDCR is initialized to 00_H by a reset.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TDE
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 27.6 DSADCCTDCR Register Contents

Bit Position	Bit Name	Function
7 to 1	—	Reserved These bits are always read as 0. The write value should always be 0.
0	TDE	Pin Level Self-Diagnosis Enable 0: Pin level self-diagnosis is disabled. 1: Pin level self-diagnosis is enabled. Setting TDE to 1 disconnects all analog pins from the input buffer. Clearing TDE to 0 connects all analog pins to the input buffer. When TDE is set to 1, DSANkx is fixed to the level specified by ANkxLV. The route from an analog pin to the delta-sigma ADC can be diagnosed by performing A/D conversion in this state and checking the A/D converted value.

Note 1. k = 0 to 7; x = P, N

CAUTION

To prevent malfunction, DSADCCTDCR should be set while ADACT for all delta-sigma ADCs is 0 (before starting A/D conversion) and STTRGE for all delta-sigma ADCs is 0.

27.4.4 DSADCCTDLVR — Pin Level Self-Diagnosis Level Setting Register

DSADCCTDLVR is a 16-bit readable/writable register to specify the level of pin-level self-diagnosis. DSADCCTDLVR is initialized to 0000_H by a reset.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	AN7NLV	AN7PLV	AN6NLV	AN6PLV	AN5NLV	AN5PLV	AN4NLV	AN4PLV	AN3NLV	AN3PLV	AN2NLV	AN2PLV	AN1NLV	AN1PLV	AN0NLV	AN0PLV
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 27.7 DSADCCTDLVR Register Contents

Bit Position	Bit Name	Function
15 to 0	ANkxLV	Pin Level Self-Diagnosis Level 0: Discharges (ADSVSS) DSANkx. 1: Charges (ADSVCC) DSANkx.

Note 1. k = 0 to 7; x = P, N

CAUTION

To prevent malfunction, DSADCCTDLVR should be set while ADACT for all delta-sigma ADCs is 0 (before starting A/D conversion) and STTRGE for all delta-sigma ADCs is 0.

27.5 Delta-Sigma ADC-Specific Registers

The following describes specific registers of the individual delta-sigma ADCs (DSADC0 to DSADC7).

27.5.1 DSADCmCCR — Channel Control Register (m = 0 to 7)

DSADCmCCR is a 32-bit readable/writable register to control channels. DSADCmCCR is initialized to 0000 0000_H by a reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DFENT	—	DFTAG[3:0]				—	ULS	—	—	—	DSDFTYP	—	—	GAIN[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R/W	R/W	R/W	R/W	R/W	R	R/W	R	R	R	R/W	R	R	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DFMT[3:0]				—	—	CNVCLS[1:0]		—	—	—	—	GCTRL[3:0]			
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 27.8 DSADCmCCR Register Contents (1/2)

Bit Position	Bit Name	Function
31	DFENT	DFE Entry 0: Not entered 1: Entered This bit selects whether to perform entry to the DFE (digital filter engine).
30	—	Reserved This bit is always read as 0. The write value should always be 0.
29 to 26	DFTAG[3:0]	DFE-TAG These bits are used to perform entry to a channel of the DFE for which TAG identical to the setting of DFTAG[3:0] is set. If TAG matches the setting DFTAG[3:0] for multiple channels, entry to multiple channels is performed.
25	—	Reserved This bit is always read as 0. The write value should always be 0.
24	ULS	Upper-Limit/Lower-Limit Table Select 0: Upper-limit /lower-limit check is not performed. 1: Upper-limit /lower-limit check is performed by DSADCmULLMTBR. When the A/D converted value is stored in DR, upper-limit and lower-limit values are checked using the upper-limit/lower-limit table.
23 to 21	—	Reserved These bits are always read as 0. The write value should always be 0.
20	DSDFTYP	Delta-Sigma ADC Digital Filter Type This bit specifies the digital filter type of delta-sigma ADC. This bit is valid when Fovs = 8 Msps. 0: The digital filter type depends on the DSADCmFCR setting. Set the DSADCmFCR register. 1: Fs = 1.6 Msps; ENOB = 8 bits (usage example 4) Set the DSADCmFCR register to the initial value. (Fs: Sampling rate, ENOB = Valid bits) **CAUTION** ENOB is a value when gain ×1 and differential input are used.
19, 18	—	Reserved These bits are always read as 0. The write value should always be 0.

Table 27.8 DSADCmCCR Register Contents (2/2)

Bit Position	Bit Name	Function
17, 16	GAIN[1:0]	Gain These bits specify the input gain. 0_H: ×1 1_H: ×2 2_H: ×4 3_H: ×8
15 to 12	DFMT[3:0]	Data Format These bits specify the lower bits to be zero-masked (rounded in the -∞ direction). 0_H: Not masked 1_H: Lower 1 bit masked 2_H: Lower 2 bits masked 3_H: Lower 3 bits masked 4_H: Lower 4 bits masked 5_H: Lower 5 bits masked 6_H: Lower 6 bits masked 7_H: Lower 7 bits masked 8_H: Lower 8 bits masked Others: Setting prohibited The setting of these bits is the format of data for DR, DFE, and APA. For details of data format, see **Section 27.5.2, DSADCmDIR — Data Supplementary Information Register (m = 0 to 7)**.
11, 10	—	Reserved These bits are always read as 0. The write value should always be 0.
9, 8	CNVCLS[1:0]	Conversion Type 0_H: Single-ended input, common = ADSVREFL 1_H: Single-ended input, common = ADSVREFH/2 2_H: Differential input 3_H: Self-diagnosis
7 to 4	—	Reserved These bits are always read as 0. The write value should always be 0.
3 to 0	GCTRL[3:0]	General-Purpose Control Write 0_H when self-diagnosis is not used (CNVCLS[1:0] ≠ 3_H). When self-diagnosis is used (CNVCLS = 3_H), GCTRL[3:0] are used to select the level of self-diagnosis. 7_H: ADSVREFH ×1 4_H: ADSVREFH ×1/2 0_H: ADSVREFH ×0 C_H: -ADSVREFH ×1/2 8_H: -ADSVREFH ×1 Others: Setting prohibited

CAUTIONS

1. To prevent malfunction, DSADCmCCR should be set while ADACT is 0 (before starting A/D conversion) and STTRGE is 0.
2. The filter type depends on the combination of DSADCCOSMPRCR, DSADCmCCR, and DSADCmFCR settings. Make appropriate settings to suit the purpose, as described in **Section 27.6.2, Setting Filter Type**.

27.5.2 DSADCmDIR — Data Supplementary Information Register (m = 0 to 7)

DSADCmDIR is a 32-bit read-only register to store A/D converted values and information incidental to A/D converted values. DSADCmDIR is cleared to 0000 0000_H by reading it while RDCLRE is set to 1. The WFLG bit is cleared to 0 by reading DSADCmDIR regardless of the setting of the RDCLRE bit. DSADCmDIR is initialized to 0000 0000_H by a reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	WFLG	PRTY	—	—	—	—	—			
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DR															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 27.9 DSADCmDIR Register Contents

Bit Position	Bit Name	Function
31 to 26	—	Reserved These bits are always read as 0. The write value should always be 0.
25	WFLG	Write Flag Setting condition The A/D converted value is stored in DR. Clearing condition DSADCmDIR is read.
24	PRTY	Parity Even parity bit for DR.
23 to 16	—	Reserved These bits are always read as 0. The write value should always be 0.
15 to 0	DR[15:0]	These bits store A/D converted values. For the data format, see **(1), Bits 15 to 0: Data format of DR (data register)**.

(1) Bits 15 to 0: Data format of DR (data register)

AD converted values are stored in DR. **Table 27.10** shows the DR format. The same format applies to data to be transferred to the DFE (digital filter engine) and the APA (autonomous pulse adapter).

Table 27.10 Signed Fixed-Point Format

DFMT[3:0]	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0_H	S															
1_H	S															0
2_H	S														0	0
3_H	S													0	0	0
4_H	S												0	0	0	0
5_H	S											0	0	0	0	0
6_H	S										0	0	0	0	0	0
7_H	S									0	0	0	0	0	0	0
8_H	S								0	0	0	0	0	0	0	0

↑
Decimal point position

S	: Sign bit
0	: Always 0

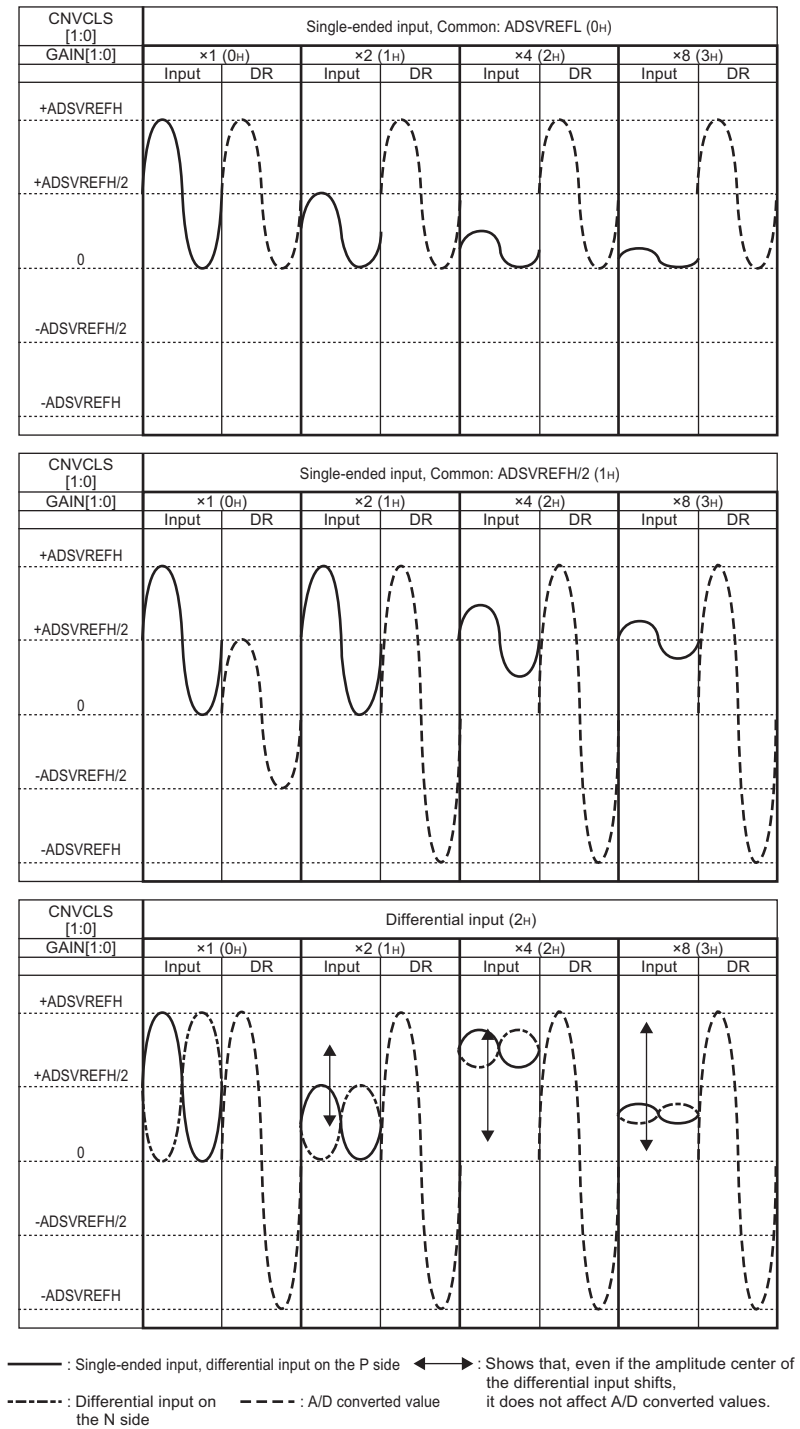


Figure 27.2 Input Range and Conversion Result Range at Each Gain

27.5.3 DSADCmADSTCR — AD Start Control Register (m = 0 to 7)

DSADCmADSTCR is an 8-bit write-only register to control starting delta-sigma ADCs. This register is always read as 0.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ADST
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 27.11 DSADCmADSTCR Register Contents

Bit Position	Bit Name	Function
7 to 1	—	Reserved These bits are always read as 0. The write value should always be 0.
0	ADST	A/D Conversion Start Condition for starting A/D conversion by ADST Writing 1 to ADST when ADACT = 0 While ADACT = 1, writing 1 to this bit is ignored and A/D conversion in progress continues. Writing 0 to this bit is ignored.

Note: It is recommended that ADST be written to while ENDTRGE is 0 considering external trigger.

27.5.4 DSADCmADENDCR — AD Stop Control Register (m = 0 to 7)

DSADCmADENDCR is an 8-bit write-only register to control stopping delta-sigma ADCs. This register is always read as 0.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ADEND
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 27.12 DSADCmADENDCR Register Contents

Bit Position	Bit Name	Function
7 to 1	—	Reserved These bits are always read as 0. The write value should always be 0.
0	ADEND	A/D Conversion Stop Condition for stopping A/D conversion by ADEND Writing 1 to ADEND when ADACT = 1 Writing 1 to this bit is ignored while ADACT = 0. Writing 0 to this bit is ignored.

Note: It is recommended that ADEND be written to while STTRGE is 0, taking external trigger.

27.5.5 DSADCmADCR — AD Control Register (m = 0 to 7)

DSADCmADCR is an 8-bit readable/writable register to control delta-sigma ADCs. DSADCmADCR is initialized to 00_H by a reset.

Bit	7	6	5	4	3	2	1	0
	—	ADSTARTE	—	ADIE	—	—	ENDTRGE	STTRGE
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R/W	R	R/W	R	R	R/W	R/W

Table 27.13 DSADCmADCR Register Contents

Bit Position	Bit Name	Function
7	—	Reserved This bit is always read as 0. The write value should always be 0.
6	ADSTARTE	AD Synchronization Start Enable 0: ADSTART is disabled. 1: ADSTART is enabled.
5	—	Reserved This bit is always read as 0. The write value should always be 0.
4	ADIE	A/D Conversion End Interrupt Enable 0: Issuing DSADIm is disabled. 1: Issuing DSADIm is enabled.
3, 2	—	Reserved These bits are always read as 0. The write value should always be 0.
1	ENDTRGE	AD End Trigger Enable 0: AD end trigger m is disabled. 1: AD end trigger m is enabled. Trigger source selection of AD end trigger m is set by the trigger selection function. For details, see **Section 25.2, Trigger Selection Function (PIC2)**.
0	STTRGE	AD Start Trigger Enable 0: AD start trigger m is disabled. 1: AD start trigger m is enabled. Trigger source selection of AD start trigger m is set by the trigger selection function. For details, see **Section 25.2, Trigger Selection Function (PIC2)**.

Note 1. m = 0 to 7

CAUTIONS

1. To prevent malfunction, ADIE in DSADCmADCR should be set while ADACT is 0 (before starting A/D conversion) and STTRGE is 0.
2. To enable module standby, ENDTRGE and STTRGE must be 0. For details, see **Section 27.7.2, Module Standby Function**.

27.5.6 DSADCmADSR — AD Status Register (m = 0 to 7)

DSADCmADSR is an 8-bit read-only register that indicates delta-sigma ADC status. DSADCmADSR is initialized to 00_H by a reset.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ADACT
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 27.14 DSADCmADSR Register Contents

Bit Position	Bit Name	Function
7 to 1	—	Reserved These bits are always read as 0. The write value should always be 0.
0	ADACT	A/D Conversion Status 0: No A/D conversion source is present. 1: An A/D conversion source is present. This bit is set when an A/D conversion start source is generated (ADACT = 1) and cleared when an A/D conversion end source is generated (ADACT = 0).

CAUTION

To enable module standby, ADACT must be 0. For details, see **Section 27.7.2, Module Standby Function**.

27.5.7 DSADCmSFTCR — Safety Control Register (m = 0 to 7)

DSADCmSFTCR is an 8-bit readable/writable register associated with safety control.

DSADCmSFTCR is initialized to 00_H by a reset.

Bit	7	6	5	4	3	2	1	0
	—	—	—	RDCLRE	ULEIE	OWEIE	PEIE	—
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R

Table 27.15 DSADCmSFTCR Register Contents

Bit Position	Bit Name	Function
7 to 5	—	Reserved These bits are always read as 0. The write value should always be 0.
4	RDCLRE	Read and Clear Enable 0: Reading DSADCmDIR does not clear DSADCmDIR. 1: Reading DSADCmDIR clears DSADCmDIR. **CAUTION** WFLG in DSADCmDIR is cleared by reading DSADCmDIR regardless of the RDCLRE value. To check clearing of the bit in DSADCmDIR by reading DSADCmDIR, read it after waiting for three read access cycles after DSADCmDIR is read. If DSADCmDIR is read without a wait, the value before clearing may be read. Since the relation between the bus clock frequency and the internal clock frequency is 1:2, a latency for clock synchronization will occur.
3	ULEIE	Upper-Limit /Lower-Limit Error Interrupt Enable 0: Disabled 1: Enabled
2	OWEIE	Overwrite Error Interrupt Enable 0: Disabled 1: Enabled
1	PEIE	Parity Error Interrupt Enable 0: Disabled 1: Enabled
0	—	Reserved These bits are always read as 0. The write value should always be 0.

CAUTION

To prevent malfunction, DSADCmSFTCR should be set while ADACT is 0 (before starting A/D conversion) and STTRGE is 0.

27.5.8 DSADCmULLMTBR — Upper-Limit/Lower-Limit Table Register (m = 0 to 7)

DSADCmULLMTBR is a 32-bit readable/writable register to set the upper-limit and lower-limit of A/D converted values. DSADCmULLMTBR is initialized to 7FFF 8000$_H$ by a reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ULMTB[15:0]															
Value after reset	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LLMTB[15:0]															
Value after reset	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 27.16 DSADCmULLMTBR Register Contents

Bit Position	Bit Name	Function
31 to 16	ULMTB[15:0]	Upper-Limit Table These bits specify the upper-limit of A/D converted values. When the following condition is met, ULE (upper-limit/lower-limit error) is set. ULMTB[15:0] < A/D converted value The ULMTB[15:0] format is the same as the DR format. Signed upper-limit value (2's complement) is compared. The A/D converted value is compared using the value before it is masked by DFMT[3:0].
15 to 0	LLMTB[15:0]	Lower-Limit Table These bits specify the lower-limit of A/D converted values. When the following condition is met, ULE (upper-limit/lower-limit error) is set. LLMTB[15:0] > A/D converted value The LLMTB[15:0] format is the same as the DR format. Signed lower-limit value (2's complement) is compared. The A/D converted value is compared using the value before it is masked by DFMT[3:0].

CAUTION

To prevent malfunction, DSADCmULLMTBR should be set while ADACT is 0 (before starting A/D conversion) and STTRGE is 0.

27.5.9 DSADCmECR — Error Clear Register (m = 0 to 7)

DSADCmECR is an 8-bit write-only register to control clearing errors. This register is always read as 0.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	ULEC	OWEC	PEC	—
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	W	W	W	R

Table 27.17 DSADCmECR Register Contents

Bit Position	Bit Name	Function
7 to 4	—	Reserved These bits are always read as 0. The write value should always be 0.
3	ULEC	Upper-Limit/Lower-Limit Error Clear Writing 0: Does not clear an upper-limit or lower-limit error. Writing 1: Clears an upper-limit or lower-limit error.
2	OWEC	Overwrite Error Clear Writing 0: Does not clear an overwrite error. Writing 1: Clears an overwrite error.
1	PEC	Parity Error Clear Writing 0: Does not clear a parity error. Writing 1: Clears a parity error.
0	—	Reserved These bits are always read as 0. The write value should always be 0.

27.5.10 DSADCmER — Error Register (m = 0 to 7)

DSADCmER is an 8-bit read-only register that indicates an error. DSADCmER is initialized to 00_H by a reset.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	ULE	OWE	PE	—
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 27.18 DSADCmER Register Contents

Bit Position	Bit Name	Function
7 to 4	—	Reserved These bits are always read as 0. The write value should always be 0.
3	ULE	Upper-Limit /Lower-Limit Error 0: No error occurred. 1: An error occurred. Setting condition The A/D converted value exceeds the specified range of the upper-limit/lower-limit table. Clearing condition Writing 1 to ULEC
2	OWE	Overwrite Error 0: No error occurred. 1: An error occurred. Setting condition The A/D converted value is written to DR when WFLG = 1. Clearing condition Writing 1 to OWEC
1	PE	Parity Error 0: No error occurred. 1: An error occurred. Setting condition A parity error is detected. Clearing condition Writing 1 to PEC
0	—	Reserved This bit is always read as 0. The write value should always be 0.

CAUTIONS

- Bits ULE and OWE are updated when the A/D converted value is written to DSADCmDIR. The PE bit is updated when DSADCmDIR is read.
- When an AD error interrupt request is generated, read DSADCmER in the interrupt exception handling and check bits being set to 1, and then clear them by using corresponding clear bits in DSADCmECR.

If error bits are not cleared and then an error occurs in the next A/D conversion, the following operation takes place, disabling error detection or identification.

1. When the same error (error bit = 1) occurs:
   No AD error interrupt request is sent. The relevant error bit in DSADCmER remains 1.
2. When a different error (error bit = 0) occurs:
   An AD error interrupt request is sent. The error bit set in the previous A/D conversion remains 1 and the relevant error bit in DSADCmER is set to 1.

27.5.11 DSADCmFCR — Digital Filter FIR Control Register (m = 0 to 7)

DSADCmFCR is an 8-bit readable/writable register to control the digital filter FIR. DSADCmFCR is initialized to 00_H by a reset.

Bit	7	6	5	4	3	2	1	0
	—	—	—	ORT	—	TPVSL[2:0]		
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R	R/W	R/W	R/W

Table 27.19 DSADCmFCR Register Contents

Bit Position	Bit Name	Function
7 to 5	—	Reserved These bits are always read as 0. The write value should always be 0.
4	ORT	Digital Filter FIR Type Output Rate Select This bit is invalid when TPVSL[2:0] = 0_H (FIR is not used). 0: FIR output rate is halved. 1: FIR output rate is quartered. The output rate of this IP (general filter type) is determined by the combination of the DSADCCOSMPRCR.OSMPR and DSADCmFCR.DSDFTYP settings.
3	—	Reserved This bit is always read as 0. The write value should always be 0.
2 to 0	TPVSL[2:0]	Digital Filter FIR Use and Factor Select These bits are valid when DSADCmCCR.DSDFTYP = 0_H. For the factors, see **Table 27.20**. 0_H: Digital filter FIR is not used. 1_H: Digital filter FIR is used and factor 1 is applied. 2_H: Digital filter FIR is used and factor 2 is applied. 3_H to 7_H: Setting prohibited

CAUTIONS

1. The filter type depends on the combination of DSADCCOSMPRCR, DSADCmCCR, and DSADCmFCR settings. Make appropriate settings to suit the purpose, as described in **Section 27.6.2, Setting Filter Type**.
2. To prevent malfunction, DSADCmFCR should be set while ADACT is 0 (before starting AD conversion) and STTRGE is 0.

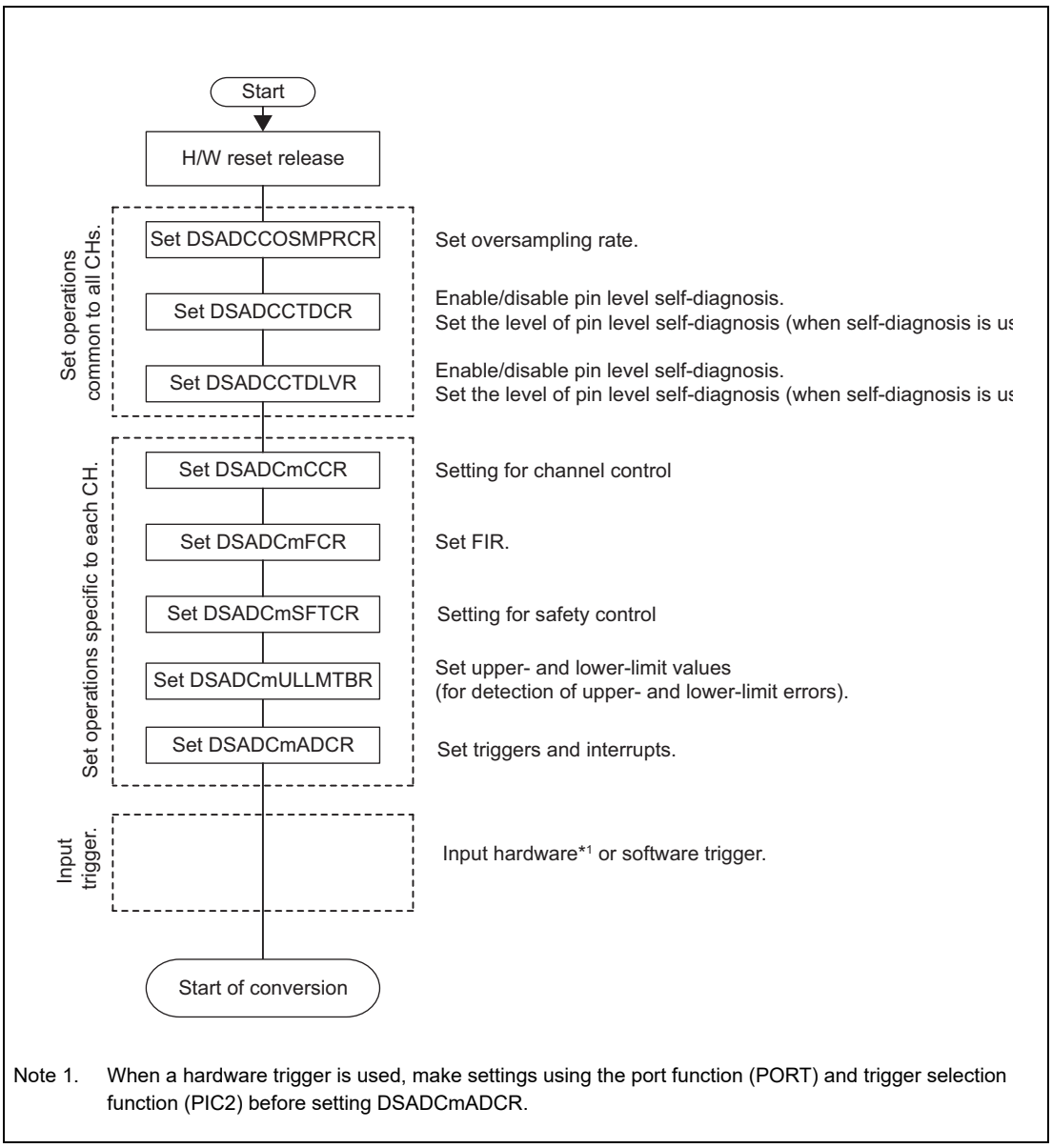
Table 27.20 Digital Filter FIR Factors

No.	Factor 1 (TPVSL = 1_H) BW = 30 kHz at 400 ksps	Factor 2 (TPVSL = 2_H) BW = 60 kHz at 400 ksps
0	681	1649
1	1940	5765
2	3911	10863
3	6089	12706
4	7674	8333
5	7890	101
6	6397	-5597
7	3581	-4689
8	417	349
9	-1978	3549
10	-2889	2155
11	-2320	-1082
12	-913	-2167
13	460	-577
14	1176	1082
15	1105	964
16	556	-170
17	-26	-694
18	-335	-298
19	-332	169
20	-210	228
21	58	-53

27.6 Operation

27.6.1 Initial Setting Flow

The delta-sigma ADC starts AD conversion by setting the registers as shown in **Figure 27.3**.



Note 1. When a hardware trigger is used, make settings using the port function (PORT) and trigger selection function (PIC2) before setting DSADCmADCR.

Figure 27.3 Initial Setting Flow

27.6.2 Setting Filter Type

The filter type depends on the combination of DSADCCOSMPRCR, DSADCmCCR, and DSADCmFCR settings. Make appropriate settings to suit the purpose, based on **Table 27.21**.

Table 27.21 Register Settings for Each Filter Type

Item	Usage Example 1	Usage Example 2	Usage Example 3	Usage Example 4
Target performance				
Fos	8 Msps	8 Msps	8 Msps	8 Msps
Fs	100 ksps	200 ksps	200 ksps	1.6 Msps
BW	30 kHz	30 kHz	60kHz	200 kHz
ENOB	13 bits	13 bits	13 bits	8 bits
Register settings				
DSADCCOSMPRCR.OSMPR	1_H	1_H	1_H	1_H
DSADCmCCR.DSDFTYP	0_H	0_H	0_H	1_H
DSADCmFCR.ORT	1_H	0_H	0_H	0_H
DSADCmFCR.TPVSL[2:0]	1_H	1_H	2_H	0_H

27.6.3 A/D Conversion Time

After the ADST bit in DSADCmADSTCR is set to 1 and then the A/D conversion start delay time (t_D) and the internal stabilization time (t_W) have passed, the delta-sigma ADC starts A/D conversion processing.

Figure 27.4 shows the A/D conversion start and end timing. **Figure 27.5** shows a budget for A/D conversion processing. The first A/D processing time (t_{AD}) includes A/D conversion start delay time (t_D), internal stabilization time (t_W), and digital filter processing delay time (t_{DF}).

The second and subsequent A/D conversion processing time (t_{AD}) is a sampling cycle (t_S). **Table 27.22** shows the A/D conversion processing time.

The following equation is used to calculate A/D conversion processing time (t_{AD}).

$$t_{AD} = t_D + t_W + t_{DF}$$

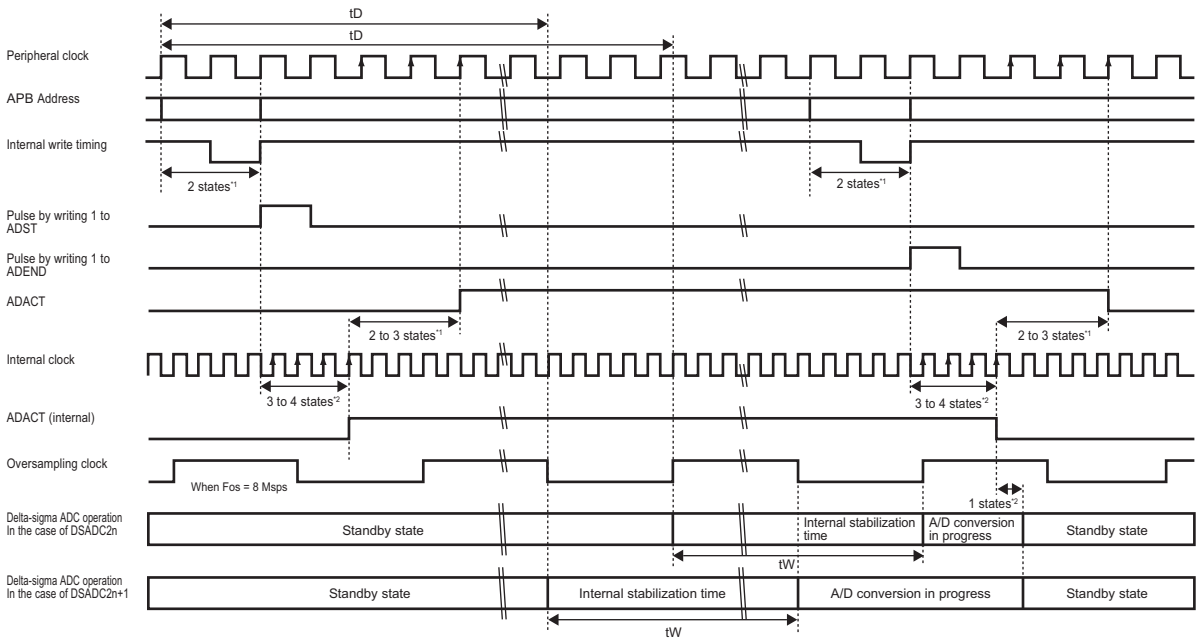
The second and subsequent t_{AD} is given by the following equation.

$$t_{AD} = t_S$$

Table 27.22 A/D Conversion Processing Time (Unit: Pϕ P-Bus Clock (40 MHz))

Item		Usage Example 1	Usage Example 2	Usage Example 3	Usage Example 4
A/D conversion processing time	t_{AD} (first time)	2595.5*1 to 2602.5*1	2595.5*1 to 2602.5*1	2595.5*1 to 2602.5*1	180.5*1 to 187.5*1
A/D conversion start delay time	t_D	16.5*1 to 23.5*1	16.5*1 to 23.5*1	16.5*1 to 23.5*1	16.5*1 to 23.5*1
Internal stabilization time	t_W	10	10	10	10
Digital filter processing delay time	t_{DF}	2569	2569	2569	154
A/D conversion processing time	t_{AD} (second or subsequent time)	400	200	200	25
Sampling cycle	t_S	400	200	200	25

Note 1. These values are for odd-numbered channels. The values for even-numbered channels are plus 2.5 bus clocks.



Note 1. Number of states of the peripheral clock

Note 2. Number of states of the internal clock

Note: n: 0 to 3, Fos: Oversampling rate

Figure 27.4 A/D Conversion Start and End

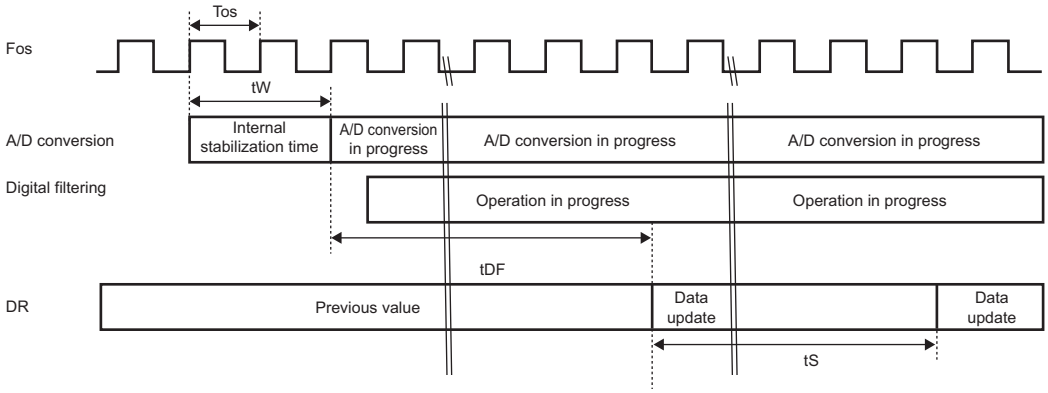


Figure 27.5 Budget for A/D Conversion Processing

27.6.4 Starting A/D Conversion by an External Trigger

DSADCm can be activated by the input of DSADTRGm. To activate DSADCm by DSADTRGm, set pin functions by the pin function controller (PFC) and select a trigger source of AD start trigger m by the trigger selection function. Input a high level to DSADTRGm and then set STTRGE in DSADCmADCR to 1. When a low level is input to DSADTRGm in this state, DSADCm detects a pulse's falling edge and sets ADACT to 1.

DSADCmADSR.ADACT is set to 1 when 3 or 4 internal clock states + 2 or 3 bus clock states have passed after the falling edge of DSADTRGm.

The timing after ADACT is set to 1 until DSADCm is activated is the same as when ADST is set to 1 by the software. For pin function settings, see **Section 2, Pins**. For selecting a trigger source of AD start trigger m, see **Section 25.2, Trigger Selection Function (PIC2)**.

To stop an active DSADCm, set ADEND in DSADCmADENDCR to 1.

27.6.5 Terminating A/D Conversion by an External Trigger

DSADCm can be deactivated by the input of DSADTRGm. To deactivate DSADCm by DSADTRGm, set pin functions by the pin function controller (PFC) and select a trigger source of AD start trigger m by the trigger selection function. Input a high level to DSADTRGm and then set ENDTRGE in DSADCmADCR to 1. When a low level is input to DSADTRGm in this state, DSADCm detects a pulse's falling edge and clears ADACT to 0 to deactivate DSADCm.

ADACT is cleared to 0 when 3 or 4 internal clock states + 2 or 3 bus clock states have passed after the falling edge of DSADTRGm.

For pin function settings, see **Section 2, Pins**. For selecting a trigger source of AD end trigger m, see **Section 25.2, Trigger Selection Function (PIC2)**.

NOTE

m = 0 to 7

27.6.6 Starting A/D Conversion by a Timer Trigger

DSADCm can be activated by a timer trigger. To activate DSADCm by a timer trigger, select a timer trigger as a trigger source of AD start trigger m by the trigger selection function. Then set STTRGE in DSADCmADCR to 1.

When the selected timer trigger is input in this state, ADACT is set to 1. The timing after the ADACT bit is set to 1 until DSADCm is activated is the same as when ADST is set to 1 by software.

For selecting a trigger source of AD start trigger m, see **Section 25.2, Trigger Selection Function (PIC2)**.

To stop an active DSADCm, set ADEND to 1.

NOTE

m = 0 to 7

27.6.7 Terminating A/D Conversion by a Timer Trigger

DSADCm can be deactivated by a timer trigger. To deactivate DSADCm by a timer trigger, select a timer trigger as a trigger source of AD end trigger m by the trigger selection function. Then set ENDTRGE in DSADCmADCR to 1.

When the selected timer trigger is input in this state, ADACT is cleared to 0 to deactivate DSADCm.

For selecting a trigger source of AD end trigger m, see **Section 25.2, Trigger Selection Function (PIC2)**.

NOTE

m = 0 to 7

27.6.8 Monitoring Using the A/D Conversion Monitor Pin

DSADENDm allows monitoring of A/D conversion of DSADCm. For pin settings, see **Section 2, Pins**.

Figure 27.6 shows an example of output from the A/D conversion monitor pin.

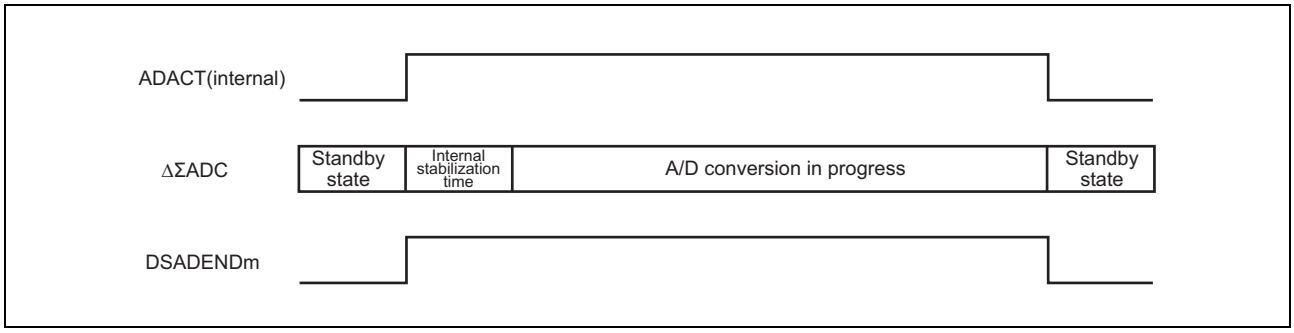


Figure 27.6 Example of A/D Conversion Monitor Pin Output

NOTE

m = 0 to 7

27.6.9 DMA Request Source

The delta-sigma ADC can generate a DMA request source (DSADIm) and send it to the DMAC to activate the DMAC (an interrupt for the INTC is not generated). Setting ADIE in DSADCmADCR to 1 outputs DSADIm at the end of the A/D conversion. Clearing ADIE to 0 disables DSADIm to be output even after the A/D conversion ends.

For settings of the DMAC, see **Section 7, DMA**.

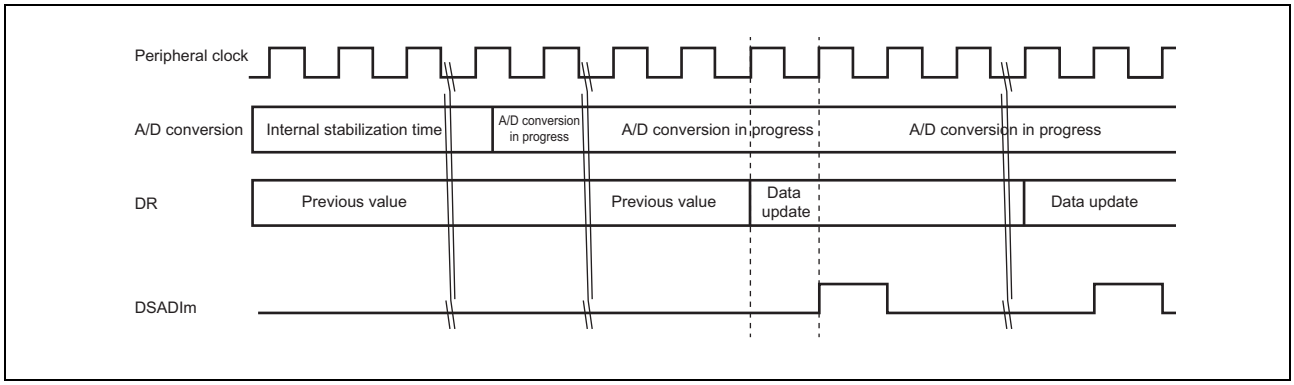


Figure 27.7 Example of DMA Request Source

NOTE

m = 0 to 7

27.6.10 AD Error Interrupt Request and AD Parity Error Notification

The delta-sigma ADC can issue an AD error interrupt request (DSADEm) to the INTC and also can issue an AD parity error notification (DSADPEm) to the error control module (ECM). The OR condition of the error source for which ULEIE and OWEIE in DSADCmSFTCR are set to 1 is issued as DSADEm. DSADEm of the error source being set to 0 can be disabled. Setting PEIE in DSADCmSFTCR to 1 enables DSADPEm, while clearing PEIE to 0 disables DSADPEm.

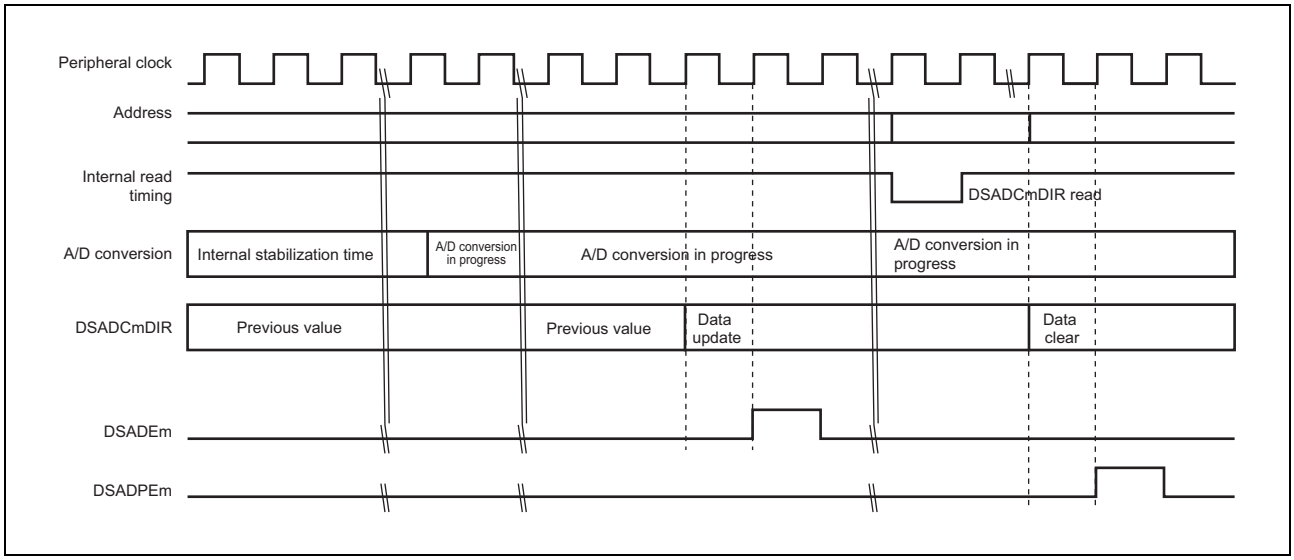


Figure 27.8 Example of Issuing of an AD Error Interrupt and AD Parity Error Notification

NOTE

m = 0 to 7

27.7 Usage Note

27.7.1 Notes on Using Analog Input Pins

Do not perform A/D conversion on the value of the same analog pin simultaneously by the delta-sigma ADC and SAR-ADC. Also, do not allow the delta-sigma ADC to perform A/D conversion on the value of the analog pin currently selected for digital input. It may lead to a deteriorated A/D accuracy.

When the delta-sigma ADC is used with single-ended input, only the analog input pins on the P side of the corresponding channel are used, and thus the analog input pins on the N side can be used for ADC input.

27.7.2 Module Standby Function

The delta-sigma ADC has the module standby function.

Module standby stops clock supply to the common units and all the channels altogether.

In addition, stop the clock for DSADC2 to 7 by setting the MSRFRAY.MS_DSAD2_7 bit of the module standby register.

To prevent malfunction, do not set module standby mode during A/D conversion or while A/D conversion start/end trigger is enabled.

- Before setting module standby mode, set the related channels as follows.
  - ADACT = 0, ENDTRGE = 0, and STTRGE = 0

Section 28 Digital Filter (DFE)

28.1 Overview

28.1.1 Features

- On-chip digital filtering (FIR/IIR) function for 16 channels
- Filter coefficients and data are stored in the on-chip RAM.
- Incorporates an accumulation circuit (for accumulation processing or decimation processing for filtered data) and a PH circuit (for peak hold processing or comparison processing).
- Generates an interrupt request when comparison calculation result is true.
- Inputs converted data from the on-chip A/D converter.

28.1.2 Overall Configuration

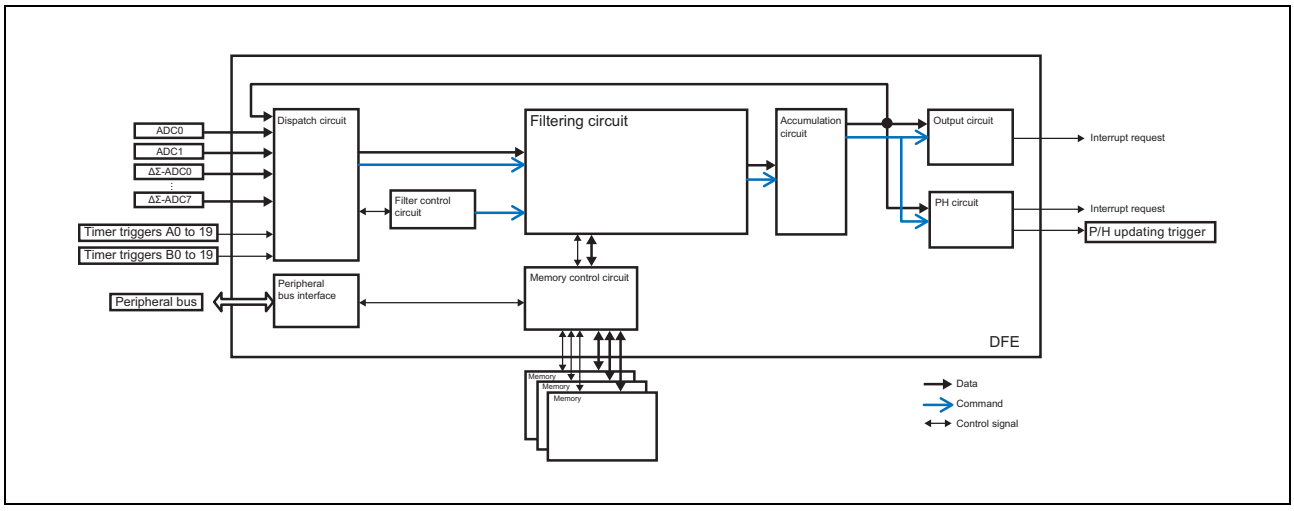


Figure 28.1 Overall Configuration of the DFE

The digital filter engine (DFE) is able to perform 16-channel filter processing on a time-division basis. FIR (up to 32 taps) or IIR (up to 6 stages) is selectable as a filter algorithm. When data is given from the A/D converter through the peripheral bus, the DFE performs FIR/IIR filter processing automatically. In normal mode, all filter processing results are stored in the output data register. In decimation mode, output data is stored in the output data register once every four filter processing, for example. When data is stored in the output data register, an interrupt request can be issued. Furthermore, the digital filter processing result can be input again to the filtering circuit.

28.1.3 Data Format

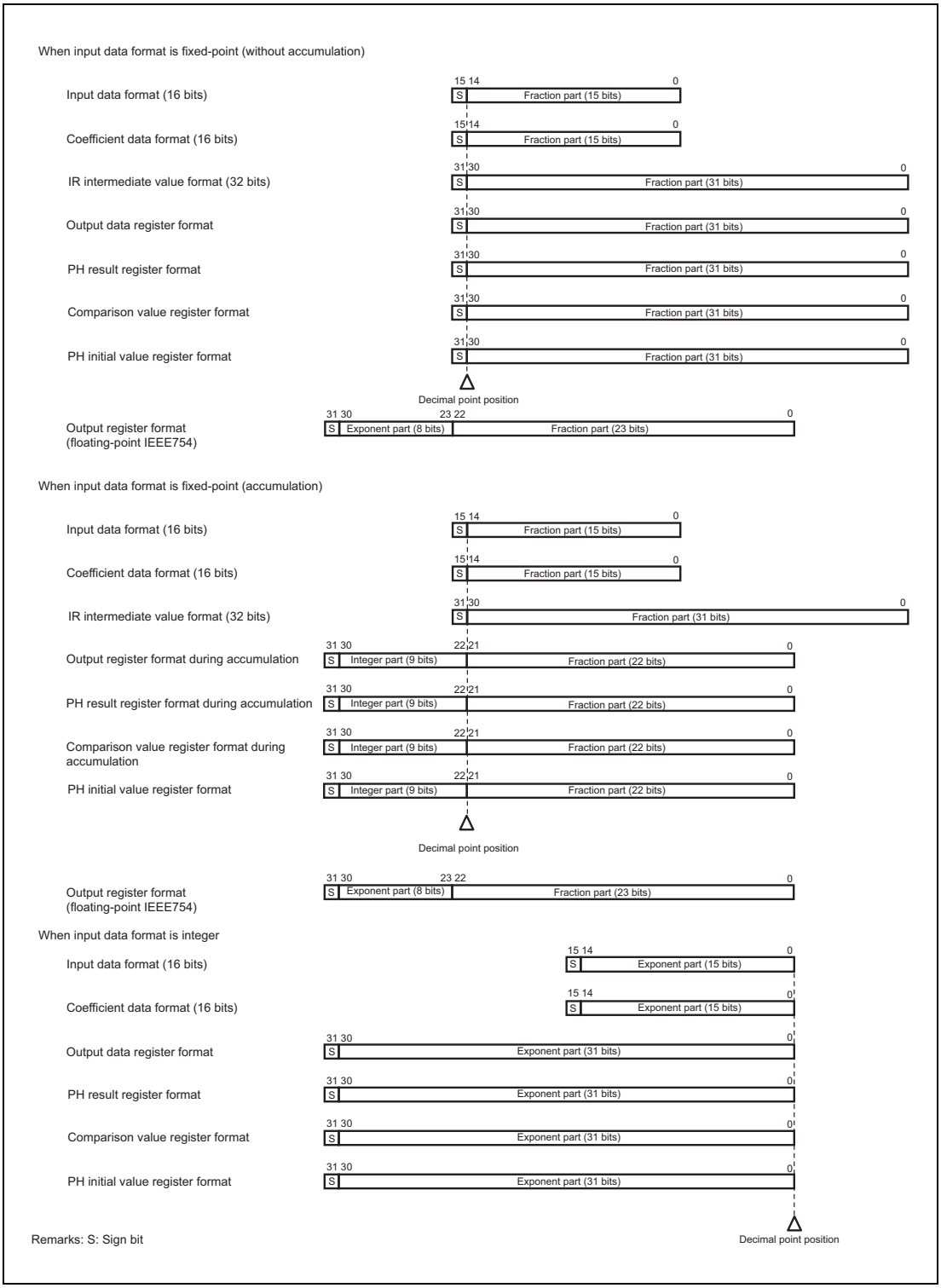


Figure 28.2 Data Format

28.1.4 Filtering Circuit

28.1.4.1 FIR Filter

- Number of taps: Selectable from 8, 16, 24, and 32
- Input data and coefficients: 16-bit signed fixed-point/16-bit integer
- Output data: 32-bit signed fixed-point/32-bit integer
- Product-sum operation accuracy: 32 bits

28.1.4.2 IIR Filter

- Number of secondary biquad stages: Selectable from 1, 2, and 3
- Input data and coefficients: 16-bit signed fixed-point data
- Intermediate data and output data: 32-bit signed fixed-point data
- Product-sum operation accuracy: 32 bits

28.1.5 Dispatch Circuit

The dispatch circuit accepts data to be filtered from the A/D converter and peripheral bus.

It assigns the accepted data to sixteen channels, then selects data to be filtered and outputs that data to the filter circuit. The data that can be input is specified for each channel. **Table 28.1** lists the data that can be input for each channel.

Table 28.1 Data that Can be Input for Each Channel

	ADC		Delta-Sigma AD Converter									
	0	1	0	1	2	3	4	5	6	7	Software	Cascade
CH0	√	√	√	√	√	×	×	×	×	×	√	√
CH1	√	√	√	√	×	√	×	×	×	×	√	√
CH2	√	√	√	√	×	×	√	×	×	×	√	√
CH3	√	√	√	√	×	×	×	√	×	×	√	√
CH4	√	√	√	√	×	×	×	×	√	×	√	√
CH5	√	√	√	√	×	×	×	×	×	√	√	√
CH6	√	√	√	√	√	×	×	×	×	×	√	√
CH7	√	√	√	√	×	√	×	×	×	×	√	√
CH8	√	√	√	√	×	×	√	×	×	×	√	√
CH9	√	√	√	√	×	×	×	√	×	×	√	√
CH10	√	√	√	√	×	×	×	×	√	×	√	√
CH11	√	√	√	√	×	×	×	×	×	√	√	√
CH12	√	√	√	√	√	×	×	×	×	×	√	√
CH13	√	√	√	√	×	√	×	×	×	×	√	√
CH14	√	√	√	√	×	×	√	×	×	×	√	√
CH15	√	√	√	√	×	×	×	√	×	×	√	√

Note: √ : Data can be input; ×: Data cannot be input

28.1.6 Accumulation Circuit

The accumulation circuit allows absolute value calculation, accumulation, and decimation processing for filtering results of all channels. Either accumulation or decimation can be specified.

- Absolute value: The absolute value of each filtering result is calculated.
- Accumulation: Accumulation of filtering results can be obtained. The maximum accumulation count is 511. Filtering results are of the S10.22 format. After accumulation processing for the predetermined number of times has been completed, one accumulation result is sent to the next processing.
- Decimation: Filtering results are decimated. If the decimation count is set to 4, for example, filtering result is output once every four filter processing. The maximum decimation count is 511.

28.1.7 PH Circuit

The PH circuit allows peak hold or comparison calculation for channels 0 to 9. The PH circuit can be linked to the accumulation circuit. For example, when accumulation/decimation is set with a specified count, peak hold processing or comparison calculation is performed on each specified count. When accumulation/decimation is not set, peak hold processing or comparison calculation is performed on each filter processing.

- Peak hold (PH): Peak hold processing of filtering results or accumulation circuit output results is possible. When the peak hold end flag is enabled, an interrupt can be requested. When the peak hold result registers are updated in the selected channels (two channels selectable), the PH update notification can be output.
- Comparison: A filtering result or accumulation circuit output result is compared with the predetermined value (=, <, >, ≥, ≤) and an interrupt can be requested according to condition match.

28.1.8 Output Circuit

In the output circuit, calculation results are stored in the output data register for each channel. When accumulation or decimation is specified, an interrupt can be requested at the same time when data is stored. In the case of input of fixed-point format data, the data format can be converted to the floating-point (IEEE754) format and the data can be stored in the output data register.

28.1.9 Terms

Table 28.2 provides definitions of terms used in this section.

Table 28.2 List of Terms (1/2)

Term	Definition in This Section
DFE operating clock	SSCG 80-MHz clock to be input to the DFE
Filtering circuit	A circuit to perform FIR filtering and IIR filtering
Data to be processed	Data for which the DFE performs filter processing, including data to be input by the AD and peripheral bus and filter processing result re-entry (cascade) data
Software input	Activation of the DFE by writing data to be processed to a register from the CPU or DMA through the peripheral bus
Input data register	A register for writing data to be processed, which is provided for 16 channels
DFE activation signal	DFE processing start signal to be input to the DFE together with data to be processed
AD tag	A tag value to be input to the DFE by the AD together with data to be processed
AD data	Data to be processed to be input to the DFE by the AD
AD input	AD tag or AD data or DFE activation signal to be input by the AD In figures, AD input shows DFE activation signal to be input by the AD.
Channel tag	A tag signal specified by the DFE's control register A, which is provided for each channel
Dispatch	Writing data to be processed to an input data register Dispatch means writing data to the input data register of the matched channel through comparison between channel tag and AD tag signal after processing is started by the DFE activation signal.
Dispatch circuit	A circuit to output filtering target data to the dispatch processing and filtering circuit
Accumulation	Accumulation of filter processing result data in each channel for the count specified in the control register
Decimation	Decimation (thinning) of filter processing result data for the count specified in the control register
Accumulation circuit	A circuit to perform absolute value calculation, accumulation, and decimation for filter processing result data
PH peak hold	Processing to hold the maximum of output data from the accumulation circuit
PH circuit	A circuit to perform PH processing and comparison processing
PH result register	A register that retains the PH processing result value
Output circuit	A circuit to write accumulation circuit output data to the output data register and to convert accumulation circuit output data to floating-point format data and write the data to the output data register
Output data register	A register used to write data after DFE processing is completed, which is provided for 16 channels
Input data overwrite error (DIOW)	An error that occurs when new data is written to the input data register while valid data is contained in the input data register (filter processing waiting state in the DFE, during filter processing, or during accumulation circuit processing)
Output data overwrite error (DOOW)	An error that occurs when new data is written to the output data register while valid data is contained in the output data register (after the DFE has written filter processing result or accumulation processing result to the output data register)
Multiplication error	An error that occurs during execution of 8000_H*8000_H in the filtering circuit
Guard error	A guard bit overflow or underflow error that occurs during execution of accumulation in the filtering circuit
Cascade error	An overflow error that occurs when 32-bit accumulation circuit output data is rounded to 16-bit data when cascade processing is enabled
Absolute value calculation error	An error to be generated when 8000_0000_H is input during calculation of absolute values in the accumulation circuit
Timer trigger	A trigger that is input from the timer.
Software trigger	A trigger generated by writing 1 to the software trigger register from the CPU or DMA

Table 28.2 List of Terms (2/2)

Term	Definition in This Section
Trigger flag	Five flags: accumulation/decimation initialization flag, accumulation/decimation disable flag, PH initialization flag, PH end flag, and filter initialization flag, which are generated by the DFE at a timer trigger, software trigger, or trigger setting register (control register in the DFE)
Accumulation/ decimation initialization flag	A trigger flag that functions as an accumulation initialization flag when accumulation is specified or as a decimation initialization flag when decimation is specified, which is used to initialize the counter in the accumulation circuit.
Accumulation/ decimation disable flag	A trigger flag that functions as an accumulation disable flag when accumulation is specified or as a decimation disable flag when decimation is specified, which stops the counter in the accumulation circuit and disables accumulation processing and decimation processing without updating output data after execution
PH initialization flag	A trigger flag that is enabled when PH is specified, which is used to initialize the counter in the PH circuit.
PH end flag	A trigger flag that is enabled when PH is specified, which is a command to terminate PH processing
Filter initialization flag	A trigger flag to be performed by adding the initialization sequence to initialization for the address pointer used in filter processing and normal filter processing.
Interrupt request	An interrupt to request activation of the interrupt controller, CPU (PCU), or DMA.

28.2 Control Registers

28.2.1 List of Registers

Table 28.3 lists registers and memory addresses of the DFE. All registers are accessible in 8, 16, or 32 bits. 32-bit write access to the coefficient memory is possible, and upper 16-bit or lower 16-bit or 32-bit write access to the data memory is possible.

Table 28.3 List of Control Registers (1/4)

Register address = FFBF 0000$_H$ + <offset address>

Offset Address	Register Name	Function	Access Size	R/W	Value after Reset
0000$_H$	CTLACH0	Control register A (Channel 0)	8, 16, 32	R/W	0000 0000$_H$
0004$_H$	CTLACH1	Control register A (Channel 1)	8, 16, 32	R/W	0000 0000$_H$
0008$_H$	CTLACH2	Control register A (Channel 2)	8, 16, 32	R/W	0000 0000$_H$
000C$_H$	CTLACH3	Control register A (Channel 3)	8, 16, 32	R/W	0000 0000$_H$
0010$_H$	CTLACH4	Control register A (Channel 4)	8, 16, 32	R/W	0000 0000$_H$
0014$_H$	CTLACH5	Control register A (Channel 5)	8, 16, 32	R/W	0000 0000$_H$
0018$_H$	CTLACH6	Control register A (Channel 6)	8, 16, 32	R/W	0000 0000$_H$
001C$_H$	CTLACH7	Control register A (Channel 7)	8, 16, 32	R/W	0000 0000$_H$
0020$_H$	CTLACH8	Control register A (Channel 8)	8, 16, 32	R/W	0000 0000$_H$
0024$_H$	CTLACH9	Control register A (Channel 9)	8, 16, 32	R/W	0000 0000$_H$
0028$_H$	CTLACH10	Control register A (Channel 10)	8, 16, 32	R/W	0000 0000$_H$
002C$_H$	CTLACH11	Control register A (Channel 11)	8, 16, 32	R/W	0000 0000$_H$
0030$_H$	CTLACH12	Control register A (Channel 12)	8, 16, 32	R/W	0000 0000$_H$
0034$_H$	CTLACH13	Control register A (Channel 13)	8, 16, 32	R/W	0000 0000$_H$
0038$_H$	CTLACH14	Control register A (Channel 14)	8, 16, 32	R/W	0000 0000$_H$
003C$_H$	CTLACH15	Control register A (Channel 15)	8, 16, 32	R/W	0000 0000$_H$
0040$_H$	CTLBCH0	Control register B (Channel 0)	8, 16, 32	R/W	0000 0000$_H$
0044$_H$	CTLBCH1	Control register B (Channel 1)	8, 16, 32	R/W	0000 0000$_H$
0048$_H$	CTLBCH2	Control register B (Channel 2)	8, 16, 32	R/W	0000 0000$_H$
004C$_H$	CTLBCH3	Control register B (Channel 3)	8, 16, 32	R/W	0000 0000$_H$
0050$_H$	CTLBCH4	Control register B (Channel 4)	8, 16, 32	R/W	0000 0000$_H$
0054$_H$	CTLBCH5	Control register B (Channel 5)	8, 16, 32	R/W	0000 0000$_H$
0058$_H$	CTLBCH6	Control register B (Channel 6)	8, 16, 32	R/W	0000 0000$_H$
005C$_H$	CTLBCH7	Control register B (Channel 7)	8, 16, 32	R/W	0000 0000$_H$
0060$_H$	CTLBCH8	Control register B (Channel 8)	8, 16, 32	R/W	0000 0000$_H$
0064$_H$	CTLBCH9	Control register B (Channel 9)	8, 16, 32	R/W	0000 0000$_H$
0068$_H$	CTLBCH10	Control register B (Channel 10)	8, 16, 32	R/W	0000 0000$_H$
006C$_H$	CTLBCH11	Control register B (Channel 11)	8, 16, 32	R/W	0000 0000$_H$
0070$_H$	CTLBCH12	Control register B (Channel 12)	8, 16, 32	R/W	0000 0000$_H$
0074$_H$	CTLBCH13	Control register B (Channel 13)	8, 16, 32	R/W	0000 0000$_H$
0078$_H$	CTLBCH14	Control register B (Channel 14)	8, 16, 32	R/W	0000 0000$_H$
007C$_H$	CTLBCH15	Control register B (Channel 15)	8, 16, 32	R/W	0000 0000$_H$
0080$_H$	DOCH0	Output data register (Channel 0)	32	R	0000 0000$_H$
0084$_H$	DOCH1	Output data register (Channel 1)	32	R	0000 0000$_H$
0088$_H$	DOCH2	Output data register (Channel 2)	32	R	0000 0000$_H$

Table 28.3 List of Control Registers (2/4)

Register address = FFBF 0000$_H$ + <offset address>

Offset Address	Register Name	Function	Access Size	R/W	Value after Reset
008C$_H$	DOCH3	Output data register (Channel 3)	32	R	0000 0000$_H$
0090$_H$	DOCH4	Output data register (Channel 4)	32	R	0000 0000$_H$
0094$_H$	DOCH5	Output data register (Channel 5)	32	R	0000 0000$_H$
0098$_H$	DOCH6	Output data register (Channel 6)	32	R	0000 0000$_H$
009C$_H$	DOCH7	Output data register (Channel 7)	32	R	0000 0000$_H$
00A0$_H$	DOCH8	Output data register (Channel 8)	32	R	0000 0000$_H$
00A4$_H$	DOCH9	Output data register (Channel 9)	32	R	0000 0000$_H$
00A8$_H$	DOCH10	Output data register (Channel 10)	32	R	0000 0000$_H$
00AC$_H$	DOCH11	Output data register (Channel 11)	32	R	0000 0000$_H$
00B0$_H$	DOCH12	Output data register (Channel 12)	32	R	0000 0000$_H$
00B4$_H$	DOCH13	Output data register (Channel 13)	32	R	0000 0000$_H$
00B8$_H$	DOCH14	Output data register (Channel 14)	32	R	0000 0000$_H$
00BC$_H$	DOCH15	Output data register (Channel 15)	32	R	0000 0000$_H$
00C0$_H$	PHCH0	PH result register (Channel 0)	32	R	8000 0000$_H$
00C4$_H$	PHCH1	PH result register (Channel 1)	32	R	8000 0000$_H$
00C8$_H$	PHCH2	PH result register (Channel 2)	32	R	8000 0000$_H$
00CC$_H$	PHCH3	PH result register (Channel 3)	32	R	8000 0000$_H$
00D0$_H$	PHCH4	PH result register (Channel 4)	32	R	8000 0000$_H$
00D4$_H$	PHCH5	PH result register (Channel 5)	32	R	8000 0000$_H$
00D8$_H$	PHCH6	PH result register (Channel 6)	32	R	8000 0000$_H$
00DC$_H$	PHCH7	PH result register (Channel 7)	32	R	8000 0000$_H$
00E0$_H$	PHCH8	PH result register (Channel 8)	32	R	8000 0000$_H$
00E4$_H$	PHCH9	PH result register (Channel 9)	32	R	8000 0000$_H$
0140$_H$	STCH0	Status register (Channel 0)	8, 16	R	0000$_H$
0144$_H$	STCH1	Status register (Channel 1)	8, 16	R	0000$_H$
0148$_H$	STCH2	Status register (Channel 2)	8, 16	R	0000$_H$
014C$_H$	STCH3	Status register (Channel 3)	8, 16	R	0000$_H$
0150$_H$	STCH4	Status register (Channel 4)	8, 16	R	0000$_H$
0154$_H$	STCH5	Status register (Channel 5)	8, 16	R	0000$_H$
0158$_H$	STCH6	Status register (Channel 6)	8, 16	R	0000$_H$
015C$_H$	STCH7	Status register (Channel 7)	8, 16	R	0000$_H$
0160$_H$	STCH8	Status register (Channel 8)	8, 16	R	0000$_H$
0164$_H$	STCH9	Status register (Channel 9)	8, 16	R	0000$_H$
0168$_H$	STCH10	Status register (Channel 10)	8, 16	R	0000$_H$
016C$_H$	STCH11	Status register (Channel 11)	8, 16	R	0000$_H$
0170$_H$	STCH12	Status register (Channel 12)	8, 16	R	0000$_H$
0174$_H$	STCH13	Status register (Channel 13)	8, 16	R	0000$_H$
0178$_H$	STCH14	Status register (Channel 14)	8, 16	R	0000$_H$
017C$_H$	STCH15	Status register (Channel 15)	8, 16	R	0000$_H$
0180$_H$	CLRSTC0	Clear status register (Channel 0)	8, 16	R/W	0000$_H$
0184$_H$	CLRSTC1	Clear status register (Channel 1)	8, 16	R/W	0000$_H$
0188$_H$	CLRSTC2	Clear status register (Channel 2)	8, 16	R/W	0000$_H$

Table 28.3 List of Control Registers (3/4)

Register address = FFBF 0000$_H$ + <offset address>

Offset Address	Register Name	Function	Access Size	R/W	Value after Reset
018C$_H$	CLRSTC3	Clear status register (Channel 3)	8, 16	R/W	0000$_H$
0190$_H$	CLRSTC4	Clear status register (Channel 4)	8, 16	R/W	0000$_H$
0194$_H$	CLRSTC5	Clear status register (Channel 5)	8, 16	R/W	0000$_H$
0198$_H$	CLRSTC6	Clear status register (Channel 6)	8, 16	R/W	0000$_H$
019C$_H$	CLRSTC7	Clear status register (Channel 7)	8, 16	R/W	0000$_H$
01A0$_H$	CLRSTC8	Clear status register (Channel 8)	8, 16	R/W	0000$_H$
01A4$_H$	CLRSTC9	Clear status register (Channel 9)	8, 16	R/W	0000$_H$
01A8$_H$	CLRSTC10	Clear status register (Channel 10)	8, 16	R/W	0000$_H$
01AC$_H$	CLRSTC11	Clear status register (Channel 11)	8, 16	R/W	0000$_H$
01B0$_H$	CLRSTC12	Clear status register (Channel 12)	8, 16	R/W	0000$_H$
01B4$_H$	CLRSTC13	Clear status register (Channel 13)	8, 16	R/W	0000$_H$
01B8$_H$	CLRSTC14	Clear status register (Channel 14)	8, 16	R/W	0000$_H$
01BC$_H$	CLRSTC15	Clear status register (Channel 15)	8, 16	R/W	0000$_H$
01C0$_H$	ERMCH0	Error mask register (Channel 0)	8	R/W	00$_H$
01C4$_H$	ERMCH1	Error mask register (Channel 1)	8	R/W	00$_H$
01C8$_H$	ERMCH2	Error mask register (Channel 2)	8	R/W	00$_H$
01CC$_H$	ERMCH3	Error mask register (Channel 3)	8	R/W	00$_H$
01D0$_H$	ERMCH4	Error mask register (Channel 4)	8	R/W	00$_H$
01D4$_H$	ERMCH5	Error mask register (Channel 5)	8	R/W	00$_H$
01D8$_H$	ERMCH6	Error mask register (Channel 6)	8	R/W	00$_H$
01DC$_H$	ERMCH7	Error mask register (Channel 7)	8	R/W	00$_H$
01E0$_H$	ERMCH8	Error mask register (Channel 8)	8	R/W	00$_H$
01E4$_H$	ERMCH9	Error mask register (Channel 9)	8	R/W	00$_H$
01E8$_H$	ERMCH10	Error mask register (Channel 10)	8	R/W	00$_H$
01EC$_H$	ERMCH11	Error mask register (Channel 11)	8	R/W	00$_H$
01F0$_H$	ERMCH12	Error mask register (Channel 12)	8	R/W	00$_H$
01F4$_H$	ERMCH13	Error mask register (Channel 13)	8	R/W	00$_H$
01F8$_H$	ERMCH14	Error mask register (Channel 14)	8	R/W	00$_H$
01FC$_H$	ERMCH15	Error mask register (Channel 15)	8	R/W	00$_H$
0200$_H$	TRGCH0	Trigger setting register (Channel 0)	16, 32	R/W	0000 0000$_H$
0204$_H$	TRGCH1	Trigger setting register (Channel 1)	16, 32	R/W	0000 0000$_H$
0208$_H$	TRGCH2	Trigger setting register (Channel 2)	16, 32	R/W	0000 0000$_H$
020C$_H$	TRGCH3	Trigger setting register (Channel 3)	16, 32	R/W	0000 0000$_H$
0210$_H$	TRGCH4	Trigger setting register (Channel 4)	16, 32	R/W	0000 0000$_H$
0214$_H$	TRGCH5	Trigger setting register (Channel 5)	16, 32	R/W	0000 0000$_H$
0218$_H$	TRGCH6	Trigger setting register (Channel 6)	16, 32	R/W	0000 0000$_H$
021C$_H$	TRGCH7	Trigger setting register (Channel 7)	16, 32	R/W	0000 0000$_H$
0220$_H$	TRGCH8	Trigger setting register (Channel 8)	16, 32	R/W	0000 0000$_H$
0224$_H$	TRGCH9	Trigger setting register (Channel 9)	16, 32	R/W	0000 0000$_H$
0228$_H$	TRGCH10	Trigger setting register (Channel 10)	16, 32	R/W	0000 0000$_H$
022C$_H$	TRGCH11	Trigger setting register (Channel 11)	16, 32	R/W	0000 0000$_H$
0230$_H$	TRGCH12	Trigger setting register (Channel 12)	16, 32	R/W	0000 0000$_H$

Table 28.3 List of Control Registers (4/4)

Register address = FFBF 0000$_H$ + <offset address>

Offset Address	Register Name	Function	Access Size	R/W	Value after Reset
0234$_H$	TRGCH13	Trigger setting register (Channel 13)	16, 32	R/W	0000 0000$_H$
0238$_H$	TRGCH14	Trigger setting register (Channel 14)	16, 32	R/W	0000 0000$_H$
023C$_H$	TRGCH15	Trigger setting register (Channel 15)	16, 32	R/W	0000 0000$_H$
0240$_H$	TRHCH0	Trigger history register (Channel 0)	8	R	00$_H$
0244$_H$	TRHCH1	Trigger history register (Channel 1)	8	R	00$_H$
0248$_H$	TRHCH2	Trigger history register (Channel 2)	8	R	00$_H$
024C$_H$	TRHCH3	Trigger history register (Channel 3)	8	R	00$_H$
0250$_H$	TRHCH4	Trigger history register (Channel 4)	8	R	00$_H$
0254$_H$	TRHCH5	Trigger history register (Channel 5)	8	R	00$_H$
0258$_H$	TRHCH6	Trigger history register (Channel 6)	8	R	00$_H$
025C$_H$	TRHCH7	Trigger history register (Channel 7)	8	R	00$_H$
0260$_H$	TRHCH8	Trigger history register (Channel 8)	8	R	00$_H$
0264$_H$	TRHCH9	Trigger history register (Channel 9)	8	R	00$_H$
0268$_H$	TRHCH10	Trigger history register (Channel 10)	8	R	00$_H$
026C$_H$	TRHCH11	Trigger history register (Channel 11)	8	R	00$_H$
0270$_H$	TRHCH12	Trigger history register (Channel 12)	8	R	00$_H$
0274$_H$	TRHCH13	Trigger history register (Channel 13)	8	R	00$_H$
0278$_H$	TRHCH14	Trigger history register (Channel 14)	8	R	00$_H$
027C$_H$	TRHCH15	Trigger history register (Channel 15)	8	R	00$_H$
0280$_H$	CPA	Comparison value setting register A	32	R/W	0000 0000$_H$
0284$_H$	CPB	Comparison value setting register B	32	R/W	0000 0000$_H$
0288$_H$	CPC	Comparison value setting register C	32	R/W	0000 0000$_H$
028C$_H$	CPD	Comparison value setting register D	32	R/W	0000 0000$_H$
0290$_H$	PHIA	PH initial value setting register A	32	R/W	8000 0000$_H$
0294$_H$	PHIB	PH initial value setting register B	32	R/W	8000 0000$_H$
0298$_H$	PHIC	PH initial value setting register C	32	R/W	8000 0000$_H$
029C$_H$	PHID	PH initial value setting register D	32	R/W	8000 0000$_H$
02B0$_H$	ACA	Accumulation/decimation count setting register A	16	R/W	0000$_H$
02B4$_H$	ACB	Accumulation/decimation count setting register B	16	R/W	0000$_H$
02B8$_H$	ACC	Accumulation/decimation count setting register C	16	R/W	0000$_H$
02BC$_H$	ACD	Accumulation/decimation count setting register D	16	R/W	0000$_H$
02C0$_H$	DI	Software input data register	32	R/W	0000 0000$_H$
02C4$_H$	TRG	Software trigger register	8	R/W	00$_H$
02CC$_H$	ST	DFE status register	8, 16, 32	R	0000 0000$_H$
0300$_H$	PITRG	PH initialization/end timer trigger select register	8, 16, 32	R/W	0000 0000$_H$
0304$_H$	MITRG	Accumulation/decimation initialization/prohibition timer trigger select register	8, 16, 32	R/W	0000 0000$_H$
0308$_H$	FITRG	Filter initialization timer trigger select register	8, 16, 32	R/W	0000 0000$_H$
0310$_H$	PHUPDC0	PH update notification setting register 0	8	R/W	00$_H$
0314$_H$	PHUPDC1	PH update notification setting register 1	8	R/W	00$_H$
1000$_H$ to 13FF$_H$	—	Coefficient memory area (1 KB)	—	R/W	—
2000$_H$ to 27FF$_H$	—	Data memory area (2 KB)	—	R/W	—

28.2.2 CTLACHn — Control Register A (n = 0 to 15)

This register is used to perform filter processing, control interrupt requests, and set channel tag and cascade tag for each channel.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	CATAG				CAEN		—	—	—	—	TAG			
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CMD				—	—	—	FMT	—	IEE	IEC	IEO	—	—	—	EN
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R	R/W	R/W	R/W	R	R	R	R/W

Table 28.4 CTLACHn Register Contents (1/2)

Bit Position	Bit Name	Function
29 to 26	CATAG	Tag Value (Cascade Tag) for Reentering Filter Result (Cascade Input) 0000_B to 1111_B: These bits specify the tag value (CTLACHn.TAG) of the channel to which accumulation circuit output data is to be reentered (cascaded) during execution of cascade.
25, 24	CAEN	Cascade Enable 00_B: Cascade is disabled. Output data of the accumulation circuit is written to the output data register. (Initial value) 01_B: Setting prohibited 10_B: Cascade is enabled. Output data of the accumulation circuit is written to the output data register together with cascade. 11_B: Cascade is enabled. Output data of the accumulation circuit is not written to the output data register.
19 to 16	TAG	Channel Tag 0000_B to 1111_B: When specifying the channel for processing input data from each AD for this channel, set the same value as the AD tag value that is input from the AD. When specifying the input data processing channel when cascade is enabled, set the same value as the tag value (CTLA.CATAG) of the cascade output source. When specifying the software input data processing channel for this channel, set the same value as the tag value (DI.TAG) of software input.
15 to 12	CMD	Filter Processing Select 0000_B: FIR 8TAP (Initial value) 0001_B: FIR 16TAP 0010_B: FIR 24TAP 0011_B: FIR 32TAP 0100_B: Setting prohibited 0101_B: Setting prohibited 0110_B: Setting prohibited 0111_B: Setting prohibited 1000_B: IIR secondary biquad 1 stage 1001_B: IIR secondary biquad 2 stages 1010_B: IIR secondary biquad 3 stages 1011_B to 1111_B: Setting prohibited
8	FMT	Input Data Format Select 0: 16-bit fixed-point (Initial value) 1: 16-bit integer To perform IIR processing, accumulation processing, or floating-point processing, specify 16-bit fixed-point.

Table 28.4 CTLACHn Register Contents (2/2)

Bit Position	Bit Name	Function
6	IEE	Error Interrupt Request Enable 0: An error interrupt request is disabled. (Initial value) 1: An error interrupt request is enabled.
5	IEC	Condition Match Interrupt Request Enable 0: A PH end interrupt request is disabled when PH is specified. (Initial value) When comparison calculation is specified, an interrupt request when the comparison calculation result is true is disabled. (Initial value) 1: A PH end interrupt request is enabled when PH is specified. When comparison calculation is specified, an interrupt request when the comparison calculation result is true is enabled. **CAUTION** PH can be specified only for channels 0 to 9.
4	IEO	Output Data Interrupt Request Enable 0: An output data interrupt when calculation result is written to the output data register is disabled. (Initial value) 1: An output data interrupt when calculation result is written to the output data register is enabled.
0	EN	Channel Enable 0: Channel disabled No processing is performed in this channel. (Initial value) 1: Channel enabled Processing is performed in this channel.

28.2.3 CTLBCHn — Control Register B (n = 0 to 15)

This register is used to set floating-point conversion, absolute value conversion, accumulation/ decimation processing, and PH/comparison processing for each channel. For channels 0 to 9, all the control bits can be set. For channels 10 to 15, bits CTLBCHn.DISB, CTLBCHn.SELB2, CTLBCHn.SELB1, and CTLBCHn.PRCSB are not supported because these channels do not have the PH/comparison processing function.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	DISB	—	—	—	—	—	—	—	DISA
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W	R	R	R	R	R	R	R	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PRCSC	SELB2			SELB1		PRCSB		—	—	SELA		—	ABS	PRCSA	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R	R/W	R/W	R/W

Table 28.5 CTLBCHn Register Contents (1/3)

Bit Position	Bit Name	Function
24	DISB	PH Processing Disable (not supported by channels 10 to 15) 0: Indicates that PH processing is in progress or PH processing is enabled (Initial value) DISB is cleared to 0 when the following condition is satisfied while DISB is 1: – Reset input – The PH initialization flag is enabled by a timer trigger or a software trigger. At this time, DISB is automatically cleared to 0. – 0 is written to DISB by setting the register by the software. When DISB is 0, PH processing is started from the data that is input after CTLBCHn.PRCSB is set to 01_B (PH processing performed) and CTLACHn.EN is set to 1. 1: Indicates that PH processing is disabled DISB is set to 1 when the following condition is satisfied while DISB is 0: – The PH end flag is enabled by a timer trigger or a software trigger. At this time, DISB is automatically set to 1. – 1 is written to DISB by setting the register by the software. When DISB is 1, PH processing is not started until the PH initialization flag is enabled after CTLBCHn.PRCSB is set to 01_B (PH processing performed) and CTLACHn.EN is set to 1. **CAUTION** Writing to the DISB or DISA is prohibited while the CTLACHn.EN bit is 1 or the STCHn.VALID bit is 1.

Table 28.5 CTLBCHn Register Contents (2/3)

Bit Position	Bit Name	Function
16	DISA	Accumulation/Decimation Processing Disable 0: Indicates that accumulation/decimation processing is in progress or accumulation/decimation processing is enabled (Initial value) DISA is cleared to 0 when the following condition is satisfied while DISA is 1: – Reset input – The accumulation/decimation initialization flag is enabled by a timer trigger or a software trigger. At this time, DISA is automatically cleared to 0. – 0 is written to DISA by setting the register by the software. When DISA is 0, accumulation/decimation processing is started from the data that is input after CTLBCHn.PRCSA is set to 01_B (accumulation/decimation processing performed) and CTLACHn.EN is set to 1. 1: Indicates that accumulation/decimation processing is disabled DISA is set to 1 when the following condition is satisfied while DISA is 0: – The accumulation/decimation disable flag is enabled by a timer trigger or a software trigger. At this time, DISA is automatically set to 1. – 1 is written to DISA by setting the register by the software. When DISA is 1, accumulation/decimation processing is not started until the accumulation/decimation initialization flag is enabled after CTLBCHn.PRCSA is set to 01_B (accumulation or decimation processing performed) and CTLACHn.EN is set to 1. **CAUTION** Writing to the DISB or DISA is prohibited while the CTLACHn.EN bit is 1 or the STCHn.VALID bit is 1.
15	PRCSC	Floating-Point Conversion 0: Floating-point conversion is not performed. (Initial value) 1: Floating-point conversion is performed. Fixed-point format (1.31/10.22) is converted to floating-point (IEEE754). This bit must not be set to 1 while the FMT bit in CTLACHn is 1 (integer mode).
14 to 12	SELB2	PH Initialization Value Register Select (not supported by channels 10 to 15) When PH is specified (CTLBCHn.PRCSB = 01_B) 000_B: PHIA register value is selected for the initial value of PH processing. (Initial value) 001_B: PHIB register value is selected for the initial value of PH processing. 010_B: PHIC register value is selected for the initial value of PH processing. 011_B: PHID register value is selected for the initial value of PH processing. 100_B to 111_B: Setting prohibited Comparison Calculation Select (not supported by channels 10 to 15) When comparison is specified (CTLBCHn.PRCSB = 10_B) 000_B: “Equal to (==)” is selected. Calculation of “equal to (==)” for the comparison target register selected by CTLBCHn.SELB1 and PH circuit input data is performed. (Initial value) 001_B: “Equal to or less than (≤)” is selected. Calculation of “equal to or less than (≤)” for the comparison target register selected by CTLBCHn.SELB1 and PH circuit input data is performed. 010_B: “Equal to or more than (≥)” is selected. Calculation of “equal to or more than (≥)” for the comparison target register selected by CTLBCHn.SELB1 and PH circuit input data is performed. 011_B: “Less than (≤)” is selected. Calculation of “less than (<)” for the comparison target register selected by CTLBCHn.SELB1 and PH circuit input data is performed. 100_B: “More than (≥)” is selected. Calculation of “more than (>)” for the comparison target register selected by CTLBCHn.SELB1 and PH circuit input data is performed. 101_B to 111_B: Setting prohibited **CAUTION** PH and comparison calculation can be specified only for channels 0 to 9.

Table 28.5 CTLBCHn Register Contents (3/3)

Bit Position	Bit Name	Function
11, 10	SELB1	Comparison Target Register Select (not supported by channels 10 to 15) 00_B: CPA register value is selected for the comparison calculation target value. (Initial value) 01_B: CPB register value is selected for the comparison calculation target value. 10_B: CPC register value is selected for the comparison calculation target value. 11_B: CPD register value is selected for the comparison calculation target value.
9, 8	PRCSB	PH Circuit Processing Select (not supported by channels 10 to 15) 00_B: Neither PH processing nor comparison calculation processing is performed. (Initial value) 01_B: PH processing is performed. 10_B: Comparison calculation processing is performed. 11_B: Setting prohibited
5, 4	SELA	Accumulation/Decimation Count Register Select 00_B: ACA value is selected for the accumulation/decimation count. (Initial value) 01_B: ACB value is selected for the accumulation/decimation count. 10_B: ACC value is selected for the accumulation/decimation count. 11_B: ACD value is selected for the accumulation/decimation count.
2	ABS	Absolute Value Calculation 0: Absolute value calculation for filter result is not performed. (Initial value) 1: Absolute value calculation for filter result is performed.
1, 0	PRCSA	Accumulation Circuit Processing Select 00_B: Neither accumulation processing nor decimation processing is performed. (Initial value) 01_B: Accumulation processing is performed. 10_B: Decimation processing is performed. 11_B: Setting prohibited

28.2.4 DOCHn — Output Data Register (n = 0 to 15)

This register is used to store the calculation result for each channel after DFE processing has been completed.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	D0															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	D0															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 28.6 DOCHn Register Contents

Bit Position	Bit Name	Function
31 to 0	D0	Output Data These bits show the DFE processing calculation result.

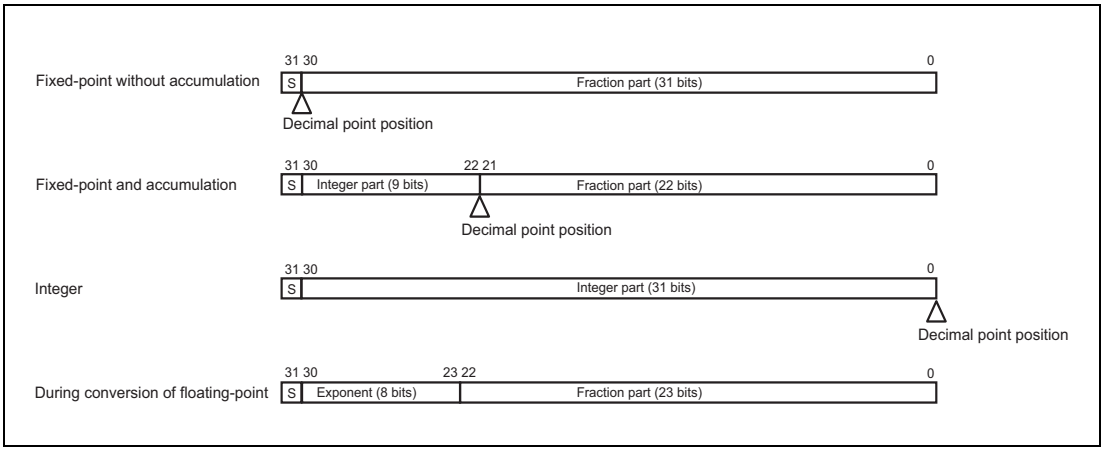


Figure 28.3 Format of Output Data Register Value

28.2.5 PHCHn — PH Result Register (n = 0 to 9)

This register is used to store the PH processing result value.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PH															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PH															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 28.7 PHCHn Register Contents

Bit Position	Bit Name	Function
31 to 0	PH	PH Result Data These bits show the PH processing calculation result.

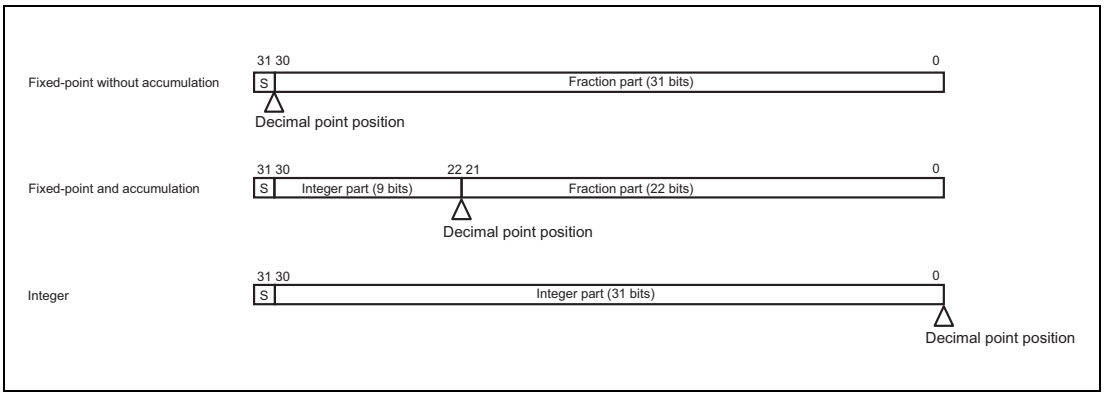


Figure 28.4 Format of PH Result Register Value

28.2.6 STCHn — Status Register (n = 0 to 15)

This register is used to indicate channel status. For channels 0 to 9, all the status bits are valid. For channels 10 to 15, STCHn.CND is not supported because these channels do not have the PH/ comparison processing function.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	VALID	—	—	CND	DOEN	—	—	CER	AER	MER	GER	DOOW	DIOW
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 28.8 STCHn Register Contents (1/2)

Bit Position	Bit Name	Function
12	VALID	Input Data Register Valid 0: Input data register value is invalid. This indicates that there is no data to be processed. (Initial value) 1: Input data register value is valid. This bit is set to 1 when data to be processed is input to the input data register, and is cleared to 0 when the data has been processed in the accumulation circuit.
9	CND	Condition Match (not supported by channels 10 to 15) When PH is specified (CTLBCHn.PRCSB = 01_B) 0: Initial value Setting CCND in the clear status register to 1 when CLRCND = 1 clears this bit to 0. 1: PH processing completed This bit is set to 1 while CND = 0 or when the PH end flag is executed. When comparison is specified (CTLBCHn.PRCSB = 10_B) 0: Initial value Setting CCND in the clear status register to 1 when CLRCND = 1 clears this bit to 0. 1: Comparison calculation result is true. This bit is set to 1 when the comparison calculation result is true while CND = 0. **CAUTION** PH and comparison calculation can be specified only for channels 0 to 9.
8	DOEN	Output Data Register Valid 0: Indicates that the output data register value is invalid. (Initial value) Setting CDOEN in the clear status register to 1 when CLRDOEN = 1 clears this bit to 0. Reading the DOCHn register when DOEN = 1 clears this bit to 0. 1: Indicates that the output data register value is valid. This bit is set to 1 when the DFE writes the calculation result data to the output data register while CTLBCHn.PRCSA is 01_B (accumulation) or CTLBCHn.PRCSA is 10_B (decimation).
5	CER	Cascade Rounding Error 0: No cascade rounding error is present. (Initial value) Setting CCER in the clear status register to 1 when CLRCER = 1 clears this bit to 0. 1: A cascade rounding error is present. This bit is set to 1 when an overflow occurs when the 32-bit accumulation circuit output data is rounded to 16-bit data when cascade is specified (CTLA.CAEN = 10_B or 11_B).

Table 28.8 STCHn Register Contents (2/2)

Bit Position	Bit Name	Function
4	AER	Absolute Value Calculation Error 0: No absolute value calculation error is present. (Initial value) Setting CAER in the clear status register to 1 when CLRAER = 1 clears this bit to 0. 1: An absolute value calculation error is present. This bit is set to 1 when the accumulation circuit input data is 8000 0000_H when absolute value calculation is specified (CTLB.ABS = 1).
3	MER	Multiplication Error 0: No multiplication error is present. (Initial value) Setting CMER in the clear status register to 1 when CLRMER = 1 clears this bit to 0. 1: A multiplication error is present. This bit is set to 1 when multiplication 8000_H*8000_H is performed in the filtering circuit.
2	GER	Guard Error 0: No guard error is present. (Initial value) Setting CGER in the clear status register to 1 when CLRGER = 1 clears this bit to 0. 1: A guard error is present. This bit is set to 1 when an overflow or underflow occurs when the accumulation result is rounded to 32 bits in the filter circuit.
1	DOOW	Output Data Overwrite Error 0: No output data overwrite error is present. (Initial value) Setting CDOOW in the clear status register to 1 when CLRDOOW = 1 clears this bit to 0. 1: An output data overwrite error is present. This bit is set to 1 when a value is written to the output data register while the output data register contains a valid value (DOEN = 1) while CTLBCHn.PRCSA is 01_B (accumulation) or CTLBCHn.PRCSA is 10_B (decimation).
0	DIOW	Input Data Overwrite Error 0: No input data overwrite is present. (Initial value) Setting CDIOW in the clear status register to 1 when CLRDIOW = 1 clears this bit to 0. 1: An input data overwrite error is present. This bit is set to 1 when a value is written to the input data register while the input data register contains a valid value (VALID = 1).

28.2.7 CLRSTCHn — Clear Status Register (n = 0 to 15)

This register is used to clear the status register. Writing 1 to each clear bit clears the corresponding bit in STCHn to 0. For channels 0 to 9, all the clear bits are valid. For channels 10 to 15, STCHn.CLRCND is not supported because these channels do not have the PH/comparison processing function.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CLRCND	CLRDOEN	—	—	CLRCER	CLRAER	CLRMER	CLRGER	CLRDOOW	CLRDIOW
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Table 28.9 CLRSTCHn Register Contents

Bit Position	Bit Name	Function
9	CLRCND	Condition Match Bit Clear (not supported by channels 10 to 15) 0: STCHn.CND is not cleared. (Initial value) 1: STCHn.CND is cleared. When STCHn.CND is cleared, this bit is automatically cleared to 0. This bit is always read as 0.
8	CLRDOEN	Output Data Enable Bit Clear 0: STCHn.DOEN is not cleared. (Initial value) 1: STCHn.DOEN is cleared. When STCHn.DOEN is cleared, this bit is automatically cleared to 0. This bit is always read as 0.
5	CLRCER	Cascade Rounding Error Bit Clear 0: STCHn.CER is not cleared. (Initial value) 1: STCHn.CER is cleared. When STCHn.CER is cleared, this bit is automatically cleared to 0. This bit is always read as 0.
4	CLRAER	Absolute Value Calculation Error Bit Clear 0: STCHn.AER is not cleared. (Initial value) 1: STCHn.AER is cleared. When STCHn.AER is cleared, this bit is automatically cleared to 0. This bit is always read as 0.
3	CLRMER	Multiplication Error Bit Clear 0: STCHn.MER is not cleared. (Initial value) 1: STCHn.MER is cleared. When STCHn.MER is cleared, this bit is automatically cleared to 0. This bit is always read as 0.
2	CLRGER	Guard Error Bit Clear 0: STCHn.GER is not cleared. (Initial value) 1: STCHn.GER is cleared. When STCHn.GER is cleared, this bit is automatically cleared to 0. This bit is always read as 0.
1	CLRDOOW	Output Data Register Overwrite Error Bit Clear 0: STCHn.DOOW is not cleared. (Initial value) 1: STCHn.DOOW is cleared. When STCHn.DOOW is cleared, this bit is automatically cleared to 0. This bit is always read as 0.
0	CLRDIOW	Input Data Register Overwrite Error Bit Clear 0: STCHn.DIOW is not cleared. (Initial value) 1: STCHn.DIOW is cleared. When STCHn.DIOW is cleared, this bit is automatically cleared to 0. This bit is always read as 0.

28.2.8 ERMCHn — Error Mask Register (n = 0 to 15)

This register is used to mask errors.

Bit	7	6	5	4	3	2	1	0
	—	—	MSKCER	MSKAER	MSKMER	MSKGER	MSKDOOW	MSKDIOW
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Table 28.10 ERMCHn Register Contents

Bit Position	Bit Name	Function
5	MSKCER	Cascade Rounding Error Bit (STCHn.CER) Mask 0: The cascade rounding error bit (STCHn.CER) is not masked. (Initial value) 1: The cascade rounding error bit (STCHn.CER) is masked.
4	MSKAER	Absolute Value Calculation Error Bit (STCHn.AER) Mask 0: The absolute value calculation error bit (STCHn.AER) is not masked. (Initial value) 1: The absolute value calculation error bit (STCHn.AER) is masked.
3	MSKMER	Multiplication Error Bit (STCHn.MER) Mask 0: The multiplication error bit (STCHn.MER) is not masked. (Initial value) 1: The multiplication error bit (STCHn.MER) is masked.
2	MSKGER	Guard Error Bit (STCHn.GER) Mask 0: The guard error bit (STCHn.GER) is not masked. (Initial value) 1: The guard error bit (STCHn.GER) is masked.
1	MSKDOOW	Output Data Overwrite Error Bit (STCHn.DOOW) Mask 0: The output data overwrite error bit (STCHn.DOOW) is not masked. (Initial value) 1: The output data overwrite error bit (STCHn.DOOW) is masked.
0	MSKDIOW	Input Data Overwrite Error Bit (STCHn.DIOW) Mask 0: The input data overwrite error bit (STCHn.DIOW) is not masked. (Initial value) 1: The input data overwrite error bit (STCHn.DIOW) is masked.

28.2.9 TRGCHn — Trigger Setting Register (n = 0 to 15)

This register is used to set generation of trigger flags by using software triggers or timer triggers. It is prohibited to set both the initialization flag and end flag as a software trigger for the PH processing. Furthermore, it is prohibited to set both the initialization flag and the disable flag as a software trigger for the accumulation/decimation processing. Any operation under prohibited settings is not guaranteed. For channels 10 to 15, bits TRGCHn.PFE, TRGCHn.PE, and TRGCHn.PT are not supported because these channels do not have the PH/comparison processing function.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	PFE		—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	AFE		—	—	PE		PT		AE		AT		FE		FT	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 28.11 TRGCHn Register Contents (1/2)

Bit Position	Bit Name	Function
19, 18	PFE	PH End Flag Trigger Setting (not supported by channels 10 to 15) 00_B: The PH end flag is not generated. (Initial value) 01_B: The PH end flag is generated by a timer trigger. 10_B: The PH end flag is generated by a software trigger. 11_B: Setting prohibited These bits disable both PH initialization flag and PH end flag to be set as a software trigger (Setting both PE and PFE bits in TRGCHn to 10_B is prohibited.) When the PH end flag is generated by a timer trigger (TRGCHn.PFE = 01_B), the PH end flag is automatically selected from among the compare-match B interrupt signals from the ATU4 which correspond to (have the same channel number in the same subblock of timer D as) the timer trigger (one of the 20 compare-match A interrupt signals from timer D of the ATU4) selected by the PH initialization flag timer trigger select bit (TRGCHn.PT).
15, 14	AFE	Accumulation/Decimation Disable Flag Trigger Setting 00_B: The accumulation/decimation disable flag is not generated. (Initial value) 01_B: The accumulation/decimation disable flag is generated by a timer trigger. 10_B: The accumulation/decimation disable flag is generated by a software trigger. 11_B: Setting prohibited These bits disable both accumulation/decimation initialization flag and accumulation/decimation disable flag to be set as a software trigger. (Setting both AE and AFE bits in TRGCHn to 10_B is prohibited.) When the accumulation/decimation disable flag is generated by a timer trigger (TRGCHn.AFE = 01_B), the accumulation/decimation disable flag is automatically selected from among the compare-match B interrupt signals from the ATU4 which correspond to (have the same channel number in the same subblock of timer D as) the timer trigger (one of the 20 compare-match A interrupt signals from timer D of the ATU4) selected by the accumulation/decimation initialization flag timer trigger select bit (TRGCHn.AT).

Table 28.11 TRGCHn Register Contents (2/2)

Bit Position	Bit Name	Function
11, 10	PE	PH Initialization Flag Trigger Setting 00_B: The PH initialization flag is not generated. (Initial value) 01_B: The PH initialization flag is generated by a timer trigger. 10_B: The PH initialization flag is generated by a software trigger. 11_B: Setting prohibited These bits disable both PH initialization flag and PH end flag to be set as a software trigger. (Setting both PE and PFE bits in TRGCHn to 10_B is prohibited.)
9, 8	PT	PH Initialization Flag and PH End Flag Timer Trigger Select (not supported by channels 10 to 15) 00_B: Timer trigger 0 is selected for generating the PH initialization flag and PH end flag. (Initial value) 01_B: Timer trigger 1 is selected for generating the PH initialization flag and PH end flag. 10_B: Timer trigger 2 is selected for generating the PH initialization flag and PH end flag. 11_B: Timer trigger 3 is selected for generating the PH initialization flag and PH end flag. These bits are valid when the PH initialization flag trigger setting bit (TRGCHn.PE) or the PH end flag trigger setting bit (TRGCHn.PFE) is 01_B. Timer triggers 0 to 3 are selected with PITMTRG0 to PITMTRG3 in PITRG.
7, 6	AE	Accumulation/Decimation Initialization Flag Trigger Setting 00_B: The accumulation/decimation initialization flag is not generated. (Initial value) 01_B: The accumulation/decimation initialization flag is generated by a timer trigger. 10_B: The accumulation/decimation initialization flag is generated by a software trigger. 11_B: Setting prohibited These bits disable both accumulation/decimation initialization flag and accumulation/decimation disable flag to be set as a software trigger. (Setting both AE and AFE bits in TRGCHn to 10_B is prohibited.)
5, 4	AT	Accumulation/Decimation Initialization Flag and Accumulation/Decimation Prohibition Flag Timer Trigger Select 00_B: Timer trigger 0 is used for generating the accumulation/decimation initialization flag and accumulation/decimation prohibition flag. (Initial value) 01_B: Timer trigger 1 is used for generating the accumulation/decimation initialization flag and accumulation/decimation prohibition flag. 10_B: Timer trigger 2 is used for generating the accumulation/decimation initialization flag and accumulation/decimation prohibition flag. 11_B: Timer trigger 3 is used for generating the accumulation/decimation initialization flag and accumulation/decimation prohibition flag. These bits are valid when the accumulation/decimation initialization flag trigger setting bit (TRGCHn.AE) or the accumulation/decimation prohibition flag trigger setting bit (TRGCHn.AFE) is 01_B. Timer triggers 0 to 3 are selected with MITMTRG0 to MITMTRG3 in MITRG.
3, 2	FE	Filter Initialization Flag Trigger Setting 00_B: The filter initialization flag is not generated. (Initial value) 01_B: The filter initialization flag is generated by a timer trigger. 10_B: The filter initialization flag is generated by a software trigger. 11_B: Setting prohibited
1, 0	FT	Filter Initialization Flag Timer Trigger Select 00_B: Timer trigger 0 is used for generating the filter initialization flag. (Initial value) 01_B: Timer trigger 1 is used for generating the filter initialization flag. 10_B: Timer trigger 2 is used for generating the filter initialization flag. 11_B: Timer trigger 3 is used for generating the filter initialization flag. Timer triggers 0 to 3 are selected with FITMTRG0 to FITMTRG3 in FITRG.

28.2.10 TRHCHn — Trigger History Register (n = 0 to 15)

This register is used to monitor execution history of software triggers, timer triggers, and trigger flags.

Bit	7	6	5	4	3	2	1	0
	—	—	—	PITS	PETS	MITS	METS	FITS
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 28.12 TRHCH Register Contents

Bit Position	Bit Name	Function
4	PITS	PH Initialization Flag Trigger History (not supported by channels 10 to 15) 0: PH initialization flag trigger is not present. (Initial value) This bit is cleared to 0 when execution of the PH initialization flag has been completed while PITS = 1. This bit is also cleared to 0 if EN in CTLACHn is set to 0 and there is no input data which is in DFE processing. 1: This bit is set to 1 when a PH initialization flag trigger occurs.
3	PETS	PH End Flag Trigger History (not supported by channels 10 to 15) 0: PH end flag trigger is not present. (Initial value) This bit is cleared to 0 when execution of the PH end flag has been completed while PETS = 1. This bit is also cleared to 0 if EN in CTLACHn is set to 0 and there is no input data which is in DFE processing. 1: This bit is set to 1 when a PH end flag trigger occurs.
2	MITS	Accumulation/Decimation Initialization Flag Trigger History 0: Accumulation/decimation initialization flag trigger is not present. (Initial value) This bit is cleared to 0 when execution of the accumulation/decimation initialization flag has been completed while MITS = 1. This bit is also cleared to 0 if EN in CTLACHn is set to 0 and there is no input data which is in DFE processing. 1: This bit is set to 1 when an accumulation/decimation initialization flag trigger occurs.
1	METS	Accumulation/Decimation Disable Flag Trigger History 0: Accumulation/decimation disable flag trigger is not present. (Initial value) This bit is cleared to 0 when execution of the accumulation/decimation disable flag has been completed while METS = 1. This bit is also cleared to 0 if EN in CTLACHn is set to 0 and there is no input data which is in DFE processing. 1: This bit is set to 1 when an accumulation/decimation disable flag trigger occurs.
0	FITS	Filter Initialization Flag Trigger History 0: Filter initialization flag trigger is not present. (Initial value) This bit is cleared to 0 when execution of the filter initialization flag has been completed while FITS = 1. This bit is also cleared to 0 if EN in CTLACHn is set to 0 and there is no input data which is in DFE processing. 1: This bit is set to 1 when a filter initialization flag trigger occurs.

28.2.11 CPA to CPD — Comparison Value Setting Registers

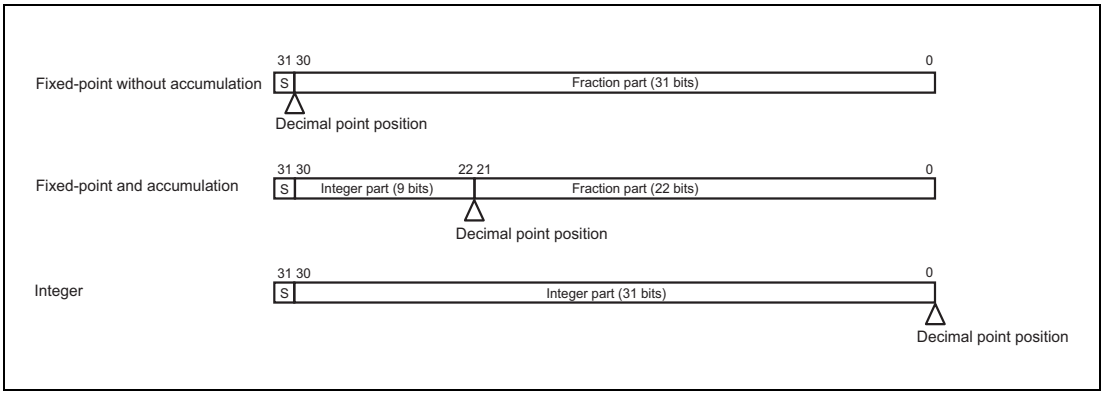
These registers are used to set a comparison calculation target value when comparison calculation is specified.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CP															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CP															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 28.13 Contents of CPA to CPD Registers

Bit Position	Bit Name	Function
31 to 0	CP	Comparison Value These bits set the comparison value when comparison calculation is specified.



28.2.12 PHIA to PHID — PH Initial Value Setting Registers

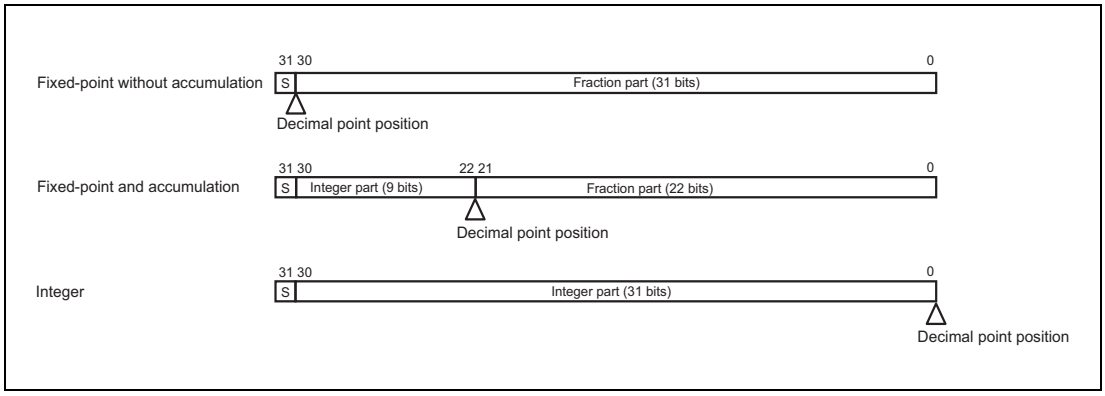
These registers are used to set the PH initial value when PH calculation is specified.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PHI															
Value after reset	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PHI															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 28.14 Contents of PHIA to PHID Registers

Bit Position	Bit Name	Function
31 to 0	PHI	PH Initial Value These bits set the PH initial value when PH calculation is specified. Initial value: Negative minimum value (8000 0000_H)



28.2.13 ACA to ACD — Accumulation/Decimation Count Setting Registers

These registers are used to set accumulation count or decimation count when accumulation/decimation is specified. A value of 0 to 511 can be set in these registers.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	AC								
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 28.15 Contents of ACA to ACD Registers

Bit Position	Bit Name	Function
8 to 0	AC	In the case of accumulation (CTLACHn.PRCSA = 01_B) Accumulation Count Setting Register 000_H: Accumulation processing is not performed.*1 (Initial value) 001_H: Accumulation is performed once. Two filter processing results are added and output from the accumulation circuit. 002_H: Accumulation is performed twice. Three filter processing results are added and output from the accumulation circuit. : $1FF_H$: Accumulation is performed 511 times. 512 filter processing results are added and output from the accumulation circuit. In the case of decimation (CTLACHn.PRCSA = 10_B) Decimation Count Setting Register 000_H: Decimation processing is not performed. (Initial value) 001_H: Decimation is performed once. Filter processing result is output once in two filter processing results. 002_H: Decimation is performed twice. Filter processing result is output once in three filter processing results. : $1FF_H$: Decimation is performed 511 times. Filter processing result is output once in 512 filter processing results.

Note 1. However, since arithmetic right shift is executed during accumulation, the format of data that is input to the accumulation circuit is (1.31) data format to (10.22) data format.

28.2.14 DI — Software Input Data Register

This register is used to input data to be processed (software input) from the CPU or DMA to the DFE through the peripheral bus. A write access to this register activates the DFE.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	TAG			
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DI															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 28.16 DI Register Contents

Bit Position	Bit Name	Function
19 to 16	TAG	Software Input Tag 0_H to F_H: Set the same value as the channel tag (CTLCHn.TAG) to be allocated.
15 to 0	DI	Software Input Data Data to be filtered is set with the input data format of the channel to be allocated. The fixed-point (1.31) format or the integer format is applicable. Set data to be filtered with the data format of the channel to be allocated. (When FMT in CTLCHn is 0, set data with the fixed-point format. When FMT in CTLCHn is 1, set data with the integer format.)

28.2.15 TRG — Software Trigger Register

This register is used to generate a software trigger by writing a value to this register from the CPU or DMA through the peripheral bus.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	TRGA
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 28.17 TRG Register Contents

Bit Position	Bit Name	Function
0	TRGA	Software Trigger Generation 0: No software trigger is generated. 1: A software trigger is generated. Writing 1 to this bit generates a software trigger. When a software trigger has occurred, this bit is automatically cleared to 0. This bit is always read as 0.

28.2.16 ST — DFE Status Register

This register is used to indicate DFE status. This register consists of logical OR of the VALID bit and each error bit in STCHn. It is supposed that the CPU uses this register to monitor error channels when the DFE outputs an error interrupt request.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	CH15S		CH14S		CH13S		CH12S		CH11S		CH10S		CH9S		CH8S	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	CH7S		CH6S		CH5S		CH4S		CH3S		CH2S		CH1S		CH0S	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 28.18 ST Register Contents

Bit Position	Bit Name	Function
bit 2*n+1-2*n	CHnS	Channel n Status ({STCHn.VALID, | (STCHn.ERR)}) 00_B: Input data register disabled with no error (Initial value) 01_B: Input data register disabled with an error 10_B: Input data register enabled with no error 11_B: Input data register enabled with an error

28.2.17 PITRG — PH Initialization/End Timer Trigger Select Register

This register selects the timer trigger 0 to 3 according to the setting of the PH initialization flag and PH end flag timer trigger select bit (TRGCHn.PT) from among 20 compare matches A and B timer D of the ATU4.

Compare match A is selected for the PH initialization flag while compare match B of the same channel is selected for the PH end flag.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	PITMTRG3					—	—	—	PITMTRG2				
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	PITMTRG1					—	—	—	PITMTRG0				
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Table 28.19 PITRG Register Contents (1/2)

Bit Position	Bit Name	Function
28 to 24	PITMTRG3	PH Initialization/End Flag Timer Trigger 3 00000_B: Uses compare match A/B (channel 0 in subblock 0) from timer D of the ATU4 as timer trigger 3 (initial value). 00001_B: Uses compare match A/B (channel 1 in subblock 0) from timer D of the ATU4 as timer trigger 3. 00010_B: Uses compare match A/B (channel 2 in subblock 0) from timer D of the ATU4 as timer trigger 3. 00011_B: Uses compare match A/B (channel 3 in subblock 0) from timer D of the ATU4 as timer trigger 3. 00100_B: Uses compare match A/B (channel 0 in subblock 2) from timer D of the ATU4 as timer trigger 3. 00101_B: Uses compare match A/B (channel 1 in subblock 2) from timer D of the ATU4 as timer trigger 3. 00110_B: Uses compare match A/B (channel 2 in subblock 2) from timer D of the ATU4 as timer trigger 3. 00111_B: Uses compare match A/B (channel 3 in subblock 2) from timer D of the ATU4 as timer trigger 3. 01000_B: Uses compare match A/B (channel 0 in subblock 4) from timer D of the ATU4 as timer trigger 3. 01001_B: Uses compare match A/B (channel 1 in subblock 4) from timer D of the ATU4 as timer trigger 3. 01010_B: Uses compare match A/B (channel 2 in subblock 4) from timer D of the ATU4 as timer trigger 3. 01011_B: Uses compare match A/B (channel 3 in subblock 4) from timer D of the ATU4 as timer trigger 3. 01100_B: Uses compare match A/B (channel 0 in subblock 6) from timer D of the ATU4 as timer trigger 3. 01101_B: Uses compare match A/B (channel 1 in subblock 6) from timer D of the ATU4 as timer trigger 3. 01110_B: Uses compare match A/B (channel 2 in subblock 6) from timer D of the ATU4 as timer trigger 3. 01111_B: Uses compare match A/B (channel 3 in subblock 6) from timer D of the ATU4 as timer trigger 3. 10000_B: Uses compare match A/B (channel 0 in subblock 9) from timer D of the ATU4 as timer trigger 3. 10001_B: Uses compare match A/B (channel 1 in subblock 9) from timer D of the ATU4 as timer trigger 3. 10010_B: Uses compare match A/B (channel 2 in subblock 9) from timer D of the ATU4 as timer trigger 3. 10011_B: Uses compare match A/B (channel 3 in subblock 9) from timer D of the ATU4 as timer trigger 3. 10100_B to 11111_B: Setting prohibited

Table 28.19 PITRG Register Contents (2/2)

Bit Position	Bit Name	Function
20 to 16	PITMTRG2	PH Initialization/End Flag Timer Trigger 2 00000_B: Uses compare match A/B (channel 0 in subblock 0) from timer D of the ATU4 as timer trigger 2 (initial value). 00001_B: Uses compare match A/B (channel 1 in subblock 0) from timer D of the ATU4 as timer trigger 2. 00010_B: Uses compare match A/B (channel 2 in subblock 0) from timer D of the ATU4 as timer trigger 2. : 10011_B: Uses compare match A/B (channel 3 in subblock 9) from timer D of the ATU4 as timer trigger 2. 10100_B to 11111_B: Setting prohibited
12 to 8	PITMTRG1	PH Initialization/End Flag Timer Trigger 1 00000_B: Uses compare match A/B (channel 0 in subblock 0) from timer D of the ATU4 as timer trigger 1 (initial value). 00001_B: Uses compare match A/B (channel 1 in subblock 0) from timer D of the ATU4 as timer trigger 1. 00010_B: Uses compare match A/B (channel 2 in subblock 0) from timer D of the ATU4 as timer trigger 1. : 10011_B: Uses compare match A/B (channel 3 in subblock 9) from timer D of the ATU4 as timer trigger 1. 10100_B to 11111_B: Setting prohibited
4 to 0	PITMTRG0	PH Initialization/End Flag Timer Trigger 0 00000_B: Uses compare match A/B (channel 0 in subblock 0) from timer D of the ATU4 as timer trigger 0 (initial value). 00001_B: Uses compare match A/B (channel 1 in subblock 0) from timer D of the ATU4 as timer trigger 0. 00010_B: Uses compare match A/B (channel 2 in subblock 0) from timer D of the ATU4 as timer trigger 0. : 10011_B: Uses compare match A/B (channel 3 in subblock 9) from timer D of the ATU4 as timer trigger 0. 10100_B to 11111_B: Setting prohibited

28.2.18 MITRG — Accumulation/Decimation Initialization/Prohibition Timer Trigger Select Register

This register selects the timer trigger 0 to 3 according to the setting of the accumulation/decimation initialization flag and the accumulation/decimation prohibition flag timer trigger select bit (TRGCHn.AT) from among 20 compare matches A and B from timer D of the ATU4.

Compare match A is selected for the accumulation/decimation initialization flag while compare match B of the same channel is selected for the accumulation/decimation prohibition flag.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	MITMTRG3					—	—	—	MITMTRG2				
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	MITMTRG1					—	—	—	MITMTRG0				
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Table 28.20 MITRG Register Contents (1/2)

Bit Position	Bit Name	Function
28 to 24	MITMTRG3	Accumulation/Decimation Initialization/Prohibition Flag Timer Trigger 3 00000_B: Uses compare match A/B (channel 0 in subblock 0) from timer D of the ATU4 as timer trigger 3 (initial value). 00001_B: Uses compare match A/B (channel 1 in subblock 0) from timer D of the ATU4 as timer trigger 3. 00010_B: Uses compare match A/B (channel 2 in subblock 0) from timer D of the ATU4 as timer trigger 3. 00011_B: Uses compare match A/B (channel 3 in subblock 0) from timer D of the ATU4 as timer trigger 3. 00100_B: Uses compare match A/B (channel 0 in subblock 2) from timer D of the ATU4 as timer trigger 3. 00101_B: Uses compare match A/B (channel 1 in subblock 2) from timer D of the ATU4 as timer trigger 3. 00110_B: Uses compare match A/B (channel 2 in subblock 2) from timer D of the ATU4 as timer trigger 3. 00111_B: Uses compare match A/B (channel 3 in subblock 2) from timer D of the ATU4 as timer trigger 3. 01000_B: Uses compare match A/B (channel 0 in subblock 4) from timer D of the ATU4 as timer trigger 3. 01001_B: Uses compare match A/B (channel 1 in subblock 4) from timer D of the ATU4 as timer trigger 3. 01010_B: Uses compare match A/B (channel 2 in subblock 4) from timer D of the ATU4 as timer trigger 3. 01011_B: Uses compare match A/B (channel 3 in subblock 4) from timer D of the ATU4 as timer trigger 3. 01100_B: Uses compare match A/B (channel 0 in subblock 6) from timer D of the ATU4 as timer trigger 3. 01101_B: Uses compare match A/B (channel 1 in subblock 6) from timer D of the ATU4 as timer trigger 3. 01110_B: Uses compare match A/B (channel 2 in subblock 6) from timer D of the ATU4 as timer trigger 3. 01111_B: Uses compare match A/B (channel 3 in subblock 6) from timer D of the ATU4 as timer trigger 3. 10000_B: Uses compare match A/B (channel 0 in subblock 9) from timer D of the ATU4 as timer trigger 3. 10001_B: Uses compare match A/B (channel 1 in subblock 9) from timer D of the ATU4 as timer trigger 3. 10010_B: Uses compare match A/B (channel 2 in subblock 9) from timer D of the ATU4 as timer trigger 3. 10011_B: Uses compare match A/B (channel 3 in subblock 9) from timer D of the ATU4 as timer trigger 3. 10100_B to 11111_B: Setting prohibited

Table 28.20 MITRG Register Contents (2/2)

Bit Position	Bit Name	Function
20 to 16	MITMTRG2	Accumulation/Decimation Initialization/Prohibition Flag Timer Trigger 2 00000_B: Uses compare match A/B (channel 0 in subblock 0) from timer D of the ATU4 as timer trigger 2 (initial value). 00001_B: Uses compare match A/B (channel 1 in subblock 0) from timer D of the ATU4 as timer trigger 2. 00010_B: Uses compare match A/B (channel 2 in subblock 0) from timer D of the ATU4 as timer trigger 2. : 10011_B: Uses compare match A/B (channel 3 in subblock 9) from timer D of the ATU4 as timer trigger 2. 10100_B to 11111_B: Setting prohibited
12 to 8	MITMTRG1	Accumulation/Decimation Initialization/Prohibition Flag Timer Trigger 1 00000_B: Uses compare match A/B (channel 0 in subblock 0) from timer D of the ATU4 as timer trigger 1 (initial value). 00001_B: Uses compare match A/B (channel 1 in subblock 0) from timer D of the ATU4 as timer trigger 1. 00010_B: Uses compare match A/B (channel 2 in subblock 0) from timer D of the ATU4 as timer trigger 1. : 10011_B: Uses compare match A/B (channel 3 in subblock 9) from timer D of the ATU4 as timer trigger 1. 10100_B to 11111_B: Setting prohibited
4 to 0	MITMTRG0	Accumulation/Decimation Initialization/Prohibition Flag Timer Trigger 0 00000_B: Uses compare match A/B (channel 0 in subblock 0) from timer D of the ATU4 as timer trigger 0 (initial value). 00001_B: Uses compare match A/B (channel 1 in subblock 0) from timer D of the ATU4 as timer trigger 0. 00010_B: Uses compare match A/B (channel 2 in subblock 0) from timer D of the ATU4 as timer trigger 0. : 10011_B: Uses compare match A/B (channel 3 in subblock 9) from timer D of the ATU4 as timer trigger 0. 10100_B to 11111_B: Setting prohibited

28.2.19 FITRG — Filter Initialization Timer Trigger Select Register

This register selects the timer trigger 0 to 3 according to the setting of the filter initialization flag timer trigger select bit (TRGCHn.FT) from among 20 compare matches A from timer D of the ATU4.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	FITMTRG3					—	—	—	FITMTRG2				
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	FITMTRG1					—	—	—	FITMTRG0				
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W	R/W	R/W	R/W	R/W

Table 28.21 FITRG Register Contents (1/2)

Bit Position	Bit Name	Function
28 to 24	FITMTRG3	Filter Initialization Flag Timer Trigger 3 00000_B: Uses compare match A (channel 0 in subblock 0) from timer D of the ATU4 as filter initialization flag timer trigger 3 (initial value). 00001_B: Uses compare match A (channel 1 in subblock 0) from timer D of the ATU4 as filter initialization flag timer trigger 3. 00010_B: Uses compare match A (channel 2 in subblock 0) from timer D of the ATU4 as filter initialization flag timer trigger 3. 00011_B: Uses compare match A (channel 3 in subblock 0) from timer D of the ATU4 as filter initialization flag timer trigger 3. 00100_B: Uses compare match A (channel 0 in subblock 2) from timer D of the ATU4 as filter initialization flag timer trigger 3. 00101_B: Uses compare match A (channel 1 in subblock 2) from timer D of the ATU4 as filter initialization flag timer trigger 3. 00110_B: Uses compare match A (channel 2 in subblock 2) from timer D of the ATU4 as filter initialization flag timer trigger 3. 00111_B: Uses compare match A (channel 3 in subblock 2) from timer D of the ATU4 as filter initialization flag timer trigger 3. 01000_B: Uses compare match A (channel 0 in subblock 4) from timer D of the ATU4 as filter initialization flag timer trigger 3. 01001_B: Uses compare match A (channel 1 in subblock 4) from timer D of the ATU4 as filter initialization flag timer trigger 3. 01010_B: Uses compare match A (channel 2 in subblock 4) from timer D of the ATU4 as filter initialization flag timer trigger 3. 01011_B: Uses compare match A (channel 3 in subblock 4) from timer D of the ATU4 as filter initialization flag timer trigger 3. 01100_B: Uses compare match A (channel 0 in subblock 6) from timer D of the ATU4 as filter initialization flag timer trigger 3. 01101_B: Uses compare match A (channel 1 in subblock 6) from timer D of the ATU4 as filter initialization flag timer trigger 3. 01110_B: Uses compare match A (channel 2 in subblock 6) from timer D of the ATU4 as filter initialization flag timer trigger 3. 01110_B: Uses compare match A (channel 3 in subblock 6) from timer D of the ATU4 as filter initialization flag timer trigger 3. 10000_B: Uses compare match A (channel 0 in subblock 9) from timer D of the ATU4 as filter initialization flag timer trigger 3. 10001_B: Uses compare match A (channel 1 in subblock 9) from timer D of the ATU4 as filter initialization flag timer trigger 3. 10010_B: Uses compare match A (channel 2 in subblock 9) from timer D of the ATU4 as filter initialization flag timer trigger 3. 10011_B: Uses compare match A (channel 3 in subblock 9) from timer D of the ATU4 as filter initialization flag timer trigger 3. 10100_B to 11111_B: Setting prohibited

Table 28.21 FITRG Register Contents (2/2)

Bit Position	Bit Name	Function
20 to 16	FITMTRG2	Filter Initialization Flag Timer Trigger 2 00000_B: Uses compare match A (channel 0 in subblock 0) from timer D of the ATU4 as filter initialization flag timer trigger 2 (initial value). 00001_B: Uses compare match A (channel 1 in subblock 0) from timer D of the ATU4 as filter initialization flag timer trigger 2. 00010_B: Uses compare match A (channel 2 in subblock 0) from timer D of the ATU4 as filter initialization flag timer trigger 2. : 10011_B: Uses compare match A (channel 3 in subblock 9) from timer D of the ATU4 as filter initialization flag timer trigger 2. 10100_B to 11111_B: Setting prohibited
12 to 8	FITMTRG1	Filter Initialization Flag Timer Trigger 1 00000_B: Uses compare match A (channel 0 in subblock 0) from timer D of the ATU4 as filter initialization flag timer trigger 1 (initial value). 00001_B: Uses compare match A (channel 1 in subblock 0) from timer D of the ATU4 as filter initialization flag timer trigger 1. 00010_B: Uses compare match A (channel 2 in subblock 0) from timer D of the ATU4 as filter initialization flag timer trigger 1. : 10011_B: Uses compare match A (channel 3 in subblock 9) from timer D of the ATU4 as filter initialization flag timer trigger 1. 10100_B to 11111_B: Setting prohibited
4 to 0	FITMTRG0	Filter Initialization Flag Timer Trigger 0 00000_B: Uses compare match A (channel 0 in subblock 0) from timer D of the ATU4 as filter initialization flag timer trigger 0 (initial value). 00001_B: Uses compare match A (channel 1 in subblock 0) from timer D of the ATU4 as filter initialization flag timer trigger 0. 00010_B: Uses compare match A (channel 2 in subblock 0) from timer D of the ATU4 as filter initialization flag timer trigger 0. : 10011_B: Uses compare match A (channel 3 in subblock 9) from timer D of the ATU4 as filter initialization flag timer trigger 0. 10100_B to 11111_B: Setting prohibited

28.2.20 PHUPDCn — PH Update Notification Setting Register n (n = 0, 1)

These registers are used to set a PH result register update notification during PH processing. One of the channels 0 to 9 can be independently selected with PHUPDC0 and PHUPDC1. The set update notification is output to the ATU4.

Bit	7	6	5	4	3	2	1	0
	PHUPDCH				—	—	—	OEPHUPD
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R	R	R	R/W

Table 28.22 PHUPDCn Register Contents

Bit Position	Bit Name	Function
7 to 4	PHUPDCH	PH Result Register Update Notification Channel Select 0_H: It is notified that PH result register has been updated in channel 0. (Initial value) 1_H: It is notified that PH result register has been updated in channel 1. 2_H: It is notified that PH result register has been updated in channel 2. : 9_H: It is notified that PH result register has been updated in channel 9. A_H to F_H: Setting prohibited
0	OEPHUPD	PH Result Register Update Notification Enable 0_B: Notification of PH result register update in the selected channel is disabled. (Initial value) 1_B: Notification of PH result register update in the selected channel is enabled.

28.3 Operation

28.3.1 Overview of Operation

The DFE imports output data from the Delta-Sigma ADC or the ADC (referred to as AD hereafter) and performs filter (FIR or IIR) processing, accumulation/decimation processing, PH/comparison calculation processing, or floating-point conversion processing for the output data.

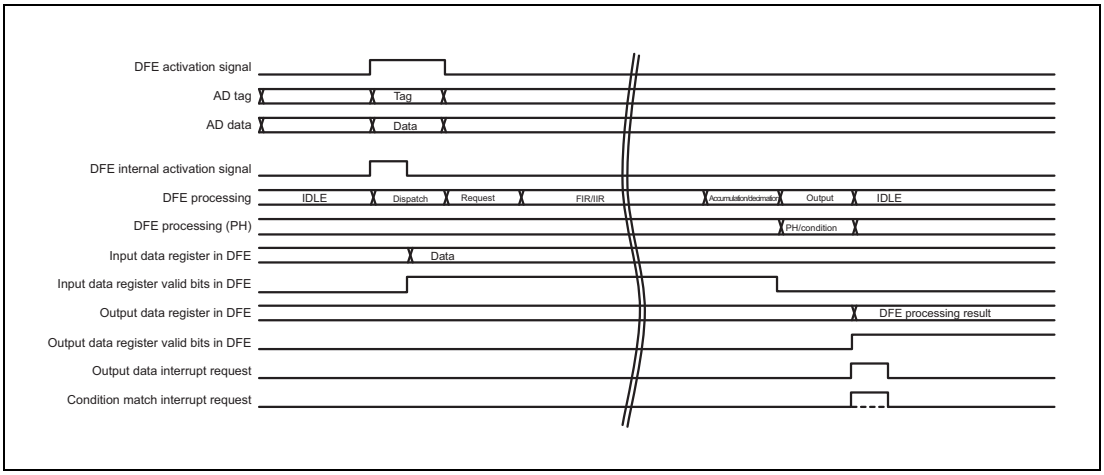


Figure 28.5 Overview of Operation

The AD inputs the DFE activation signal, AD tag, and AD conversion data (AD data) to the DFE. The DFE imports the AD data in the internal input data register to perform dispatch processing, request processing, FIR/IIR processing, accumulation/decimation processing, output processing, or PH/comparison processing, and then writes the DFE processing result to the output data register.

- Dispatch processing:
  The DFE writes data to the input data register of the relevant channel by using the AD tag from the AD and DFE's control register values. Before writing data, the input data register is enabled.
- Request processing:
  The DFE arbitrates FIR/IIR processing target data from the input data register of 16 channels and inputs the filter processing target data to the FIR/IIR processing.
- FIR/IIR processing (filter processing):
  The DFE performs FIR or IIR filter calculation. Coefficient data and intermediate values in the filter calculation are stored in the memory.
- Accumulation/decimation processing:
  The DFE performs accumulation processing or decimation processing for the FIR or IIR processing result. After the accumulation/decimation processing has been completed, the input data register is disabled.
- Output processing:
  The DFE writes the accumulation/decimation processing result to the output data register of the relevant channel. Before writing the result, the output data register is enabled and an output data interrupt request is issued.
- PH/comparison processing:
  The DFE performs PH processing or comparison processing for the accumulation/decimation processing result. In the case of PH processing, a condition match interrupt request is output when

the PH end trigger is enabled. In the case of comparison processing, a condition match interrupt request is output when the comparison calculation result is true.

28.3.2 Operating Procedures

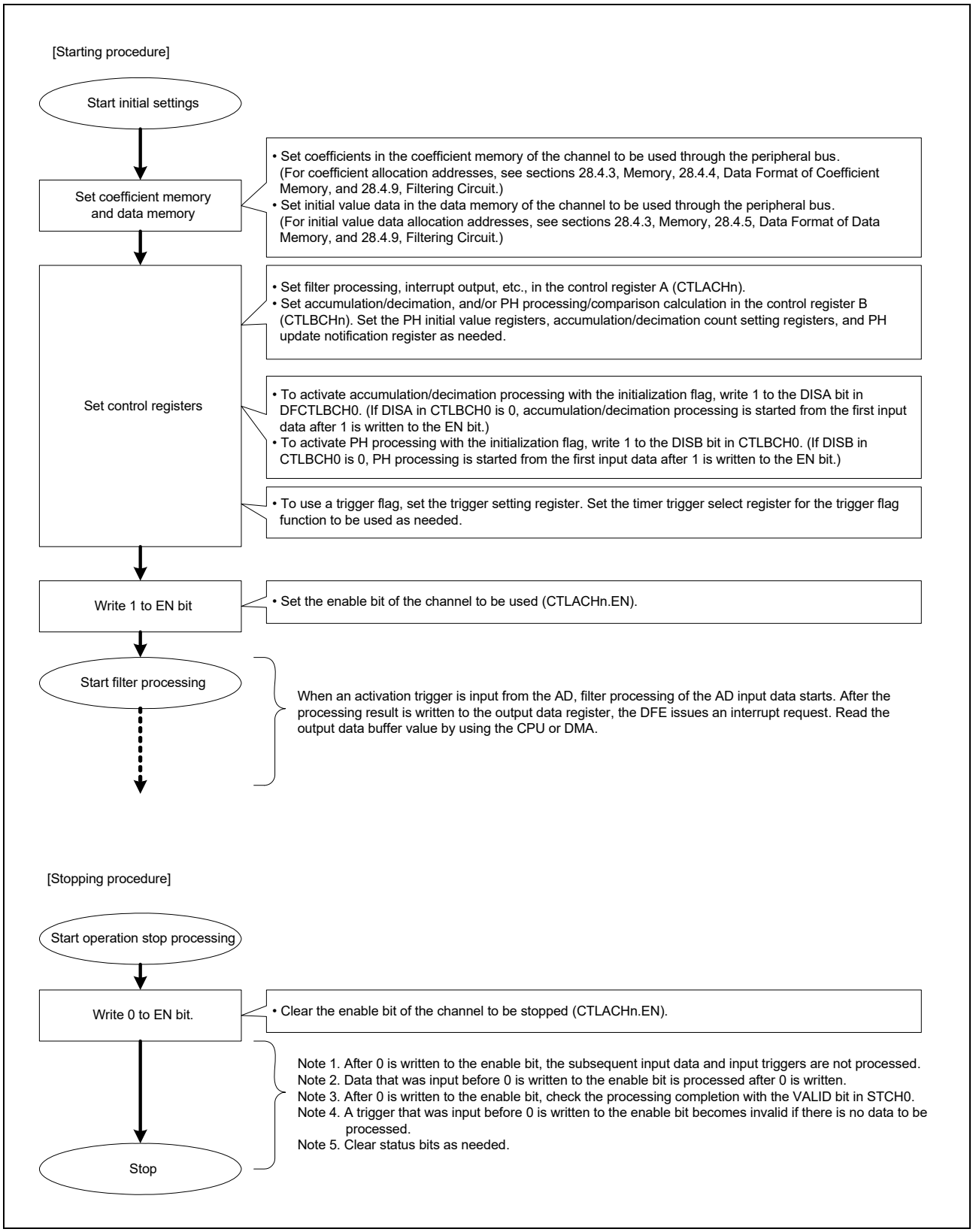


Figure 28.6 Starting and Stopping Procedures

28.3.3 Error Processing Procedure

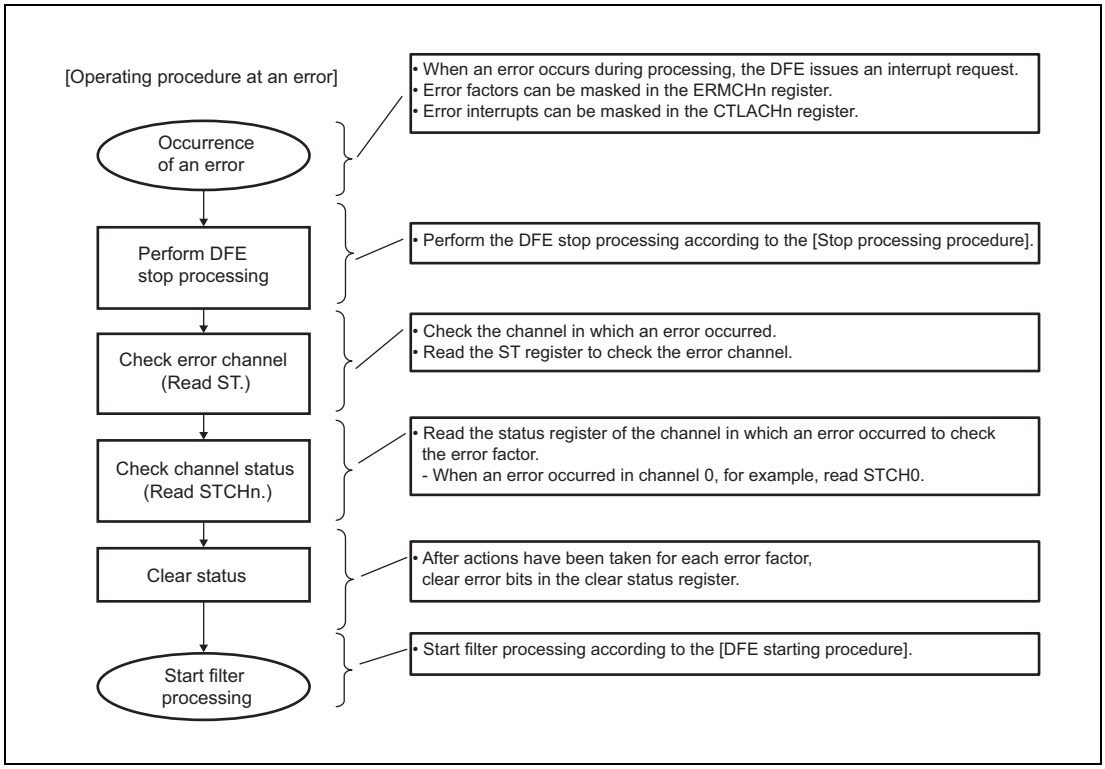


Figure 28.7 Operating Procedure when an Error Occurs

28.4 Details

28.4.1 Data Flow

Figure 28.8 shows a data flow and a block diagram of the DFE.

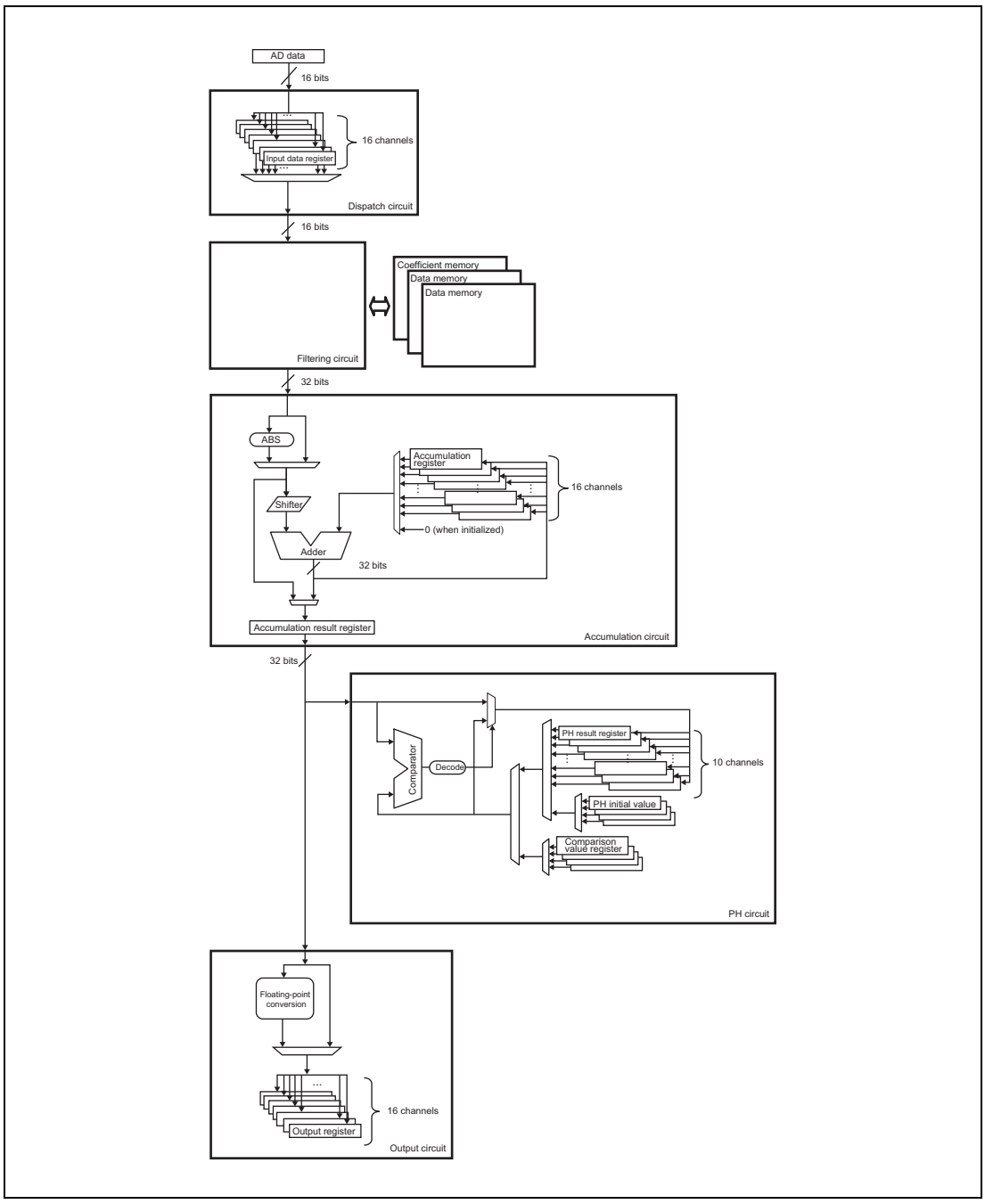


Figure 28.8 Data Flow

28.4.2 Memory Map

Figure 28.9 shows a memory map of the DFE.

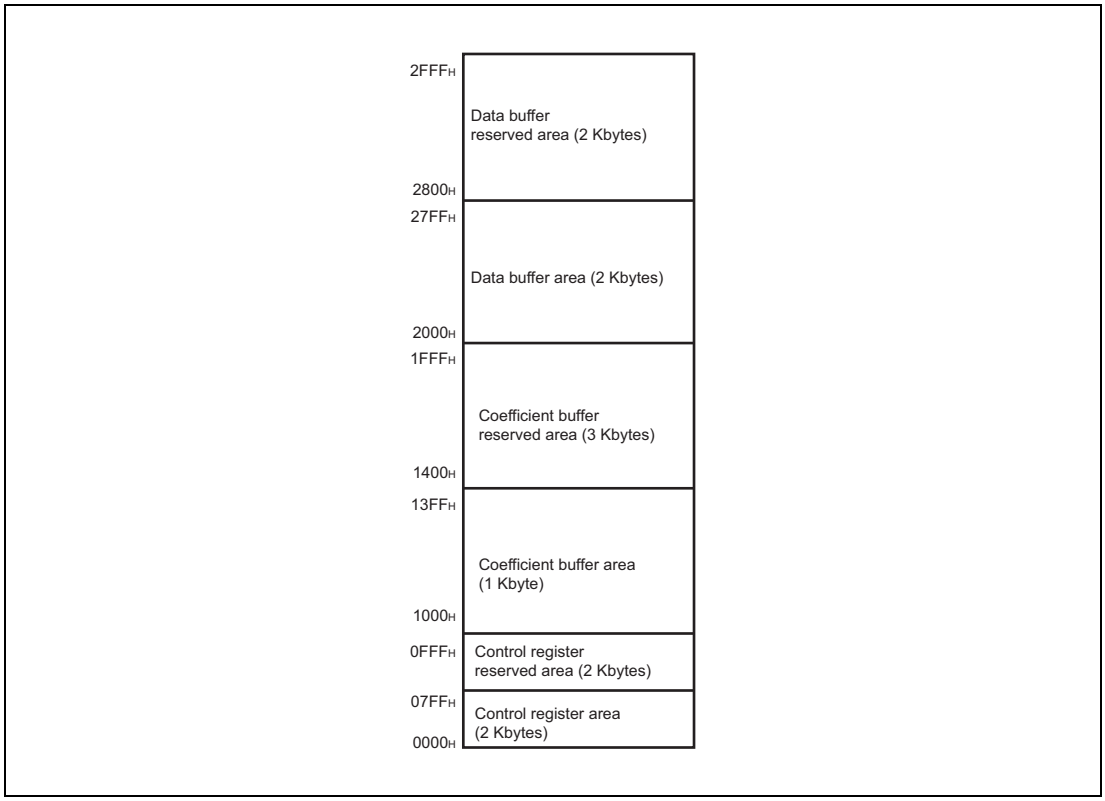


Figure 28.9 Memory Map

28.4.3 Memory

Coefficients required for filter processing are stored in the coefficient memory (CMEM), and input data used in the FIR processing, intermediate data and initial values used in the IIR processing are stored in the data memory. The coefficient memory and the data memory have an area for each channel.

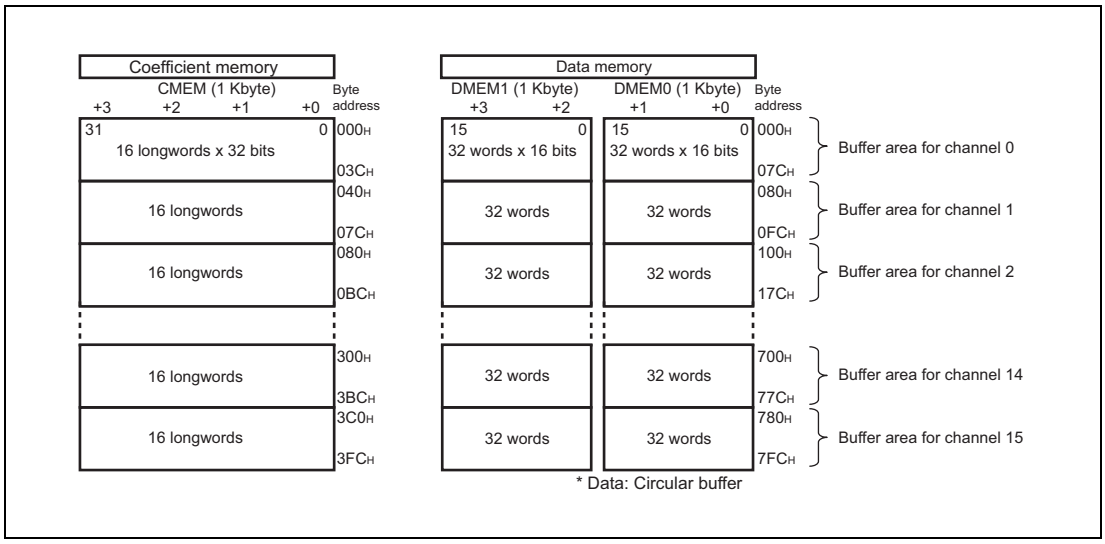


Figure 28.10 Usage of Memory

28.4.4 Data Format of Coefficient Memory

Figure 28.11 shows a data format of the coefficient memory.

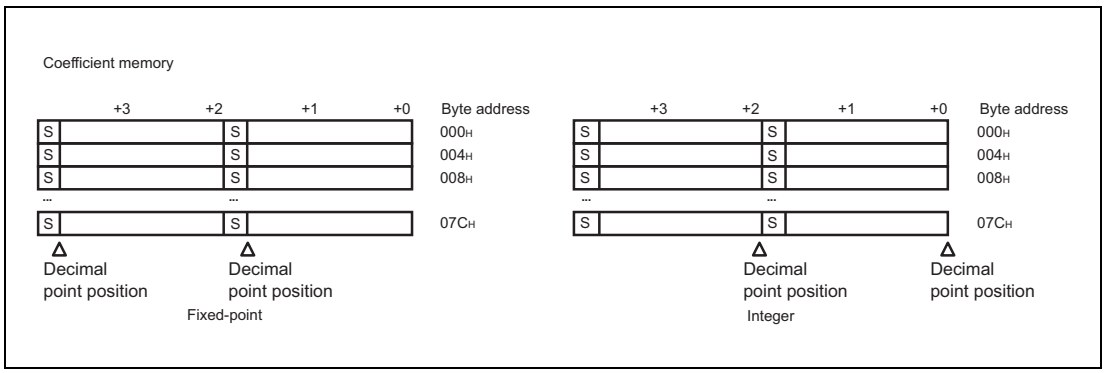


Figure 28.11 Decimal Point Positions in the Coefficient Memory

28.4.5 Data Format of Data Memory

Figure 28.12 shows decimal point positions of data in the data memory.

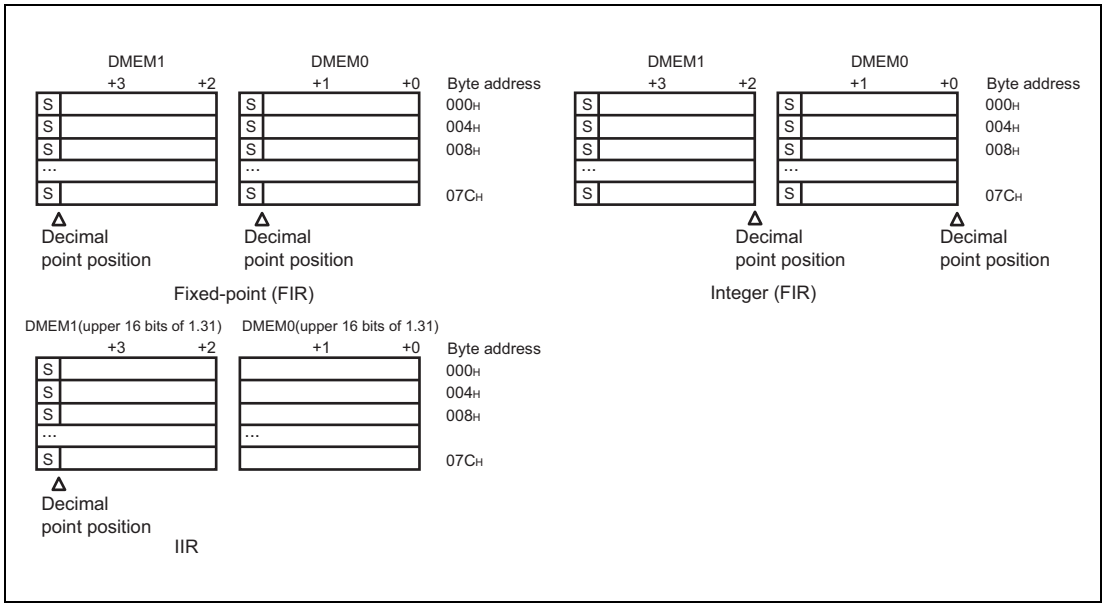


Figure 28.12 Decimal Point Positions in the Data Memory

28.4.6 Setting Control Registers

28.4.6.1 FIR (Fixed-Point)

The following table shows combinations of calculations and cascades that are selectable in the control register A (CTLACHn) and the control register B (CTLBCHn) when FIR and fixed-point format are specified in the control register A. Does not use the cascade function in channels for which accumulation is specified.

Table 28.23 FIR (Fixed-Point)

Input Format	Filter	Accumulation or Decimation	PH or Comparison	Floating-Point Conversion	Execution (Result)	Cascade (Output)*1
Fixed-point (1.15)	FIR	Accumulation	PH	Conversion specified	OK (floating-point)	NO
				Conversion not specified	OK (10.22)	NO
			Comparison	Conversion specified	OK (floating-point)	NO
				Conversion not specified	OK (10.22)	NO
			Not specified	Conversion specified	OK (floating-point)	NO
				Conversion not specified	OK (10.22)	NO
		Decimation	PH	Conversion specified	OK (floating-point)	OK (1.15)
				Conversion not specified	OK (1.31)	OK (1.15)
			Comparison	Conversion specified	OK (floating-point)	OK (1.15)
				Conversion not specified	OK (1.31)	OK (1.15)
			Not specified	Conversion specified	OK (floating-point)	OK (1.15)
				Conversion not specified	OK (1.31)	OK (1.15)
		Not specified	PH	Conversion specified	OK (floating-point)	OK (1.15)
				Conversion not specified	OK (1.31)	OK (1.15)
			Comparison	Conversion specified	OK (floating-point)	OK (1.15)
				Conversion not specified	OK (1.31)	OK (1.15)
			Not specified	Conversion specified	OK (floating-point)	OK (1.15)
				Conversion not specified	OK (1.31)	OK (1.15)

Note 1. Setting is prohibited for NO combinations. For OK combinations, set fixed-point as an input data format of cascade destination channels.

28.4.6.2 FIR (Integer)

The following table shows combinations of calculations and cascades that are selectable in the control register A (CTLACHn) and the control register B (CTLBCHn) when FIR and integer format are specified in the control register A. When integer is specified, only the FIR filter processing can be performed. Do not specify floating-point conversion.

Table 28.24 FIR (Integer)

Input Format	Filter	Accumulation or Decimation	PH or Comparison	Floating-Point Conversion	Execution (Result)*1	Cascade (Output)*2
Integer (16-bit integer)	FIR	Accumulation	PH	Conversion specified	NO	NO
				Conversion not specified	NO	NO
			Comparison	Conversion specified	NO	NO
				Conversion not specified	NO	NO
			Not specified	Conversion specified	NO	NO
				Conversion not specified	NO	NO
		Decimation	PH	Conversion specified	NO	OK (16-bit integer)
				Conversion not specified	OK (32-bit integer)	OK (16-bit integer)
			Comparison	Conversion specified	NO	OK (16-bit integer)
				Conversion not specified	OK (32-bit integer)	OK (16-bit integer)
			Not specified	Conversion specified	NO	OK (16-bit integer)
				Conversion not specified	OK (32-bit integer)	OK (16-bit integer)
		Not specified	PH	Conversion specified	NO	OK (16-bit integer)
				Conversion not specified	OK (32-bit integer)	OK (16-bit integer)
			Comparison	Conversion specified	NO	OK (16-bit integer)
				Conversion not specified	OK (32-bit integer)	OK (16-bit integer)
			Not specified	Conversion specified	NO	OK (16-bit integer)
				Conversion not specified	OK (32-bit integer)	OK (16-bit integer)

Note 1. Setting is prohibited for NO combinations.

Note 2. Setting is prohibited for NO combinations. For OK combinations, set integer as an input data format of cascade destination channels.

28.4.6.3 IIR (Fixed-Point)

The following table shows combinations of calculations and cascades that are selectable in the control register A (CTLACHn) and the control register B (CTLBCHn) when IIR and fixed-point format are specified in the control register A. When IIR is specified, only fixed-point format filter processing can be performed. When accumulation is specified, specification of cascade is prohibited.

Table 28.25 IIR

Input Format	Filter	Accumulation or Decimation	PH or Comparison	Floating-Point Conversion	Execution (Result)	Cascade (Output) *1
Fixed-point (1.15)	IIR	Accumulation	PH	Conversion specified	OK (floating-point)	NO
				Conversion not specified	OK (10.22)	NO
			Comparison	Conversion specified	OK (floating-point)	NO
				Conversion not specified	OK (10.22)	NO
			Not specified	Conversion specified	OK (floating-point)	NO
				Conversion not specified	OK (10.22)	NO
		Decimation	PH	Conversion specified	OK (floating-point)	OK (1.15)
				Conversion not specified	OK (1.31)	OK (1.15)
			Comparison	Conversion specified	OK (floating-point)	OK (1.15)
				Conversion not specified	OK (1.31)	OK (1.15)
			Not specified	Conversion specified	OK (floating-point)	OK (1.15)
				Conversion not specified	OK (1.31)	OK (1.15)
		Not specified	PH	Conversion specified	OK (floating-point)	OK (1.15)
				Conversion not specified	OK (1.31)	OK (1.15)
			Comparison	Conversion specified	OK (floating-point)	OK (1.15)
				Conversion not specified	OK (1.31)	OK (1.15)
			Not specified	Conversion specified	OK (floating-point)	OK (1.15)
				Conversion not specified	OK (1.31)	OK (1.15)

Note 1. Setting is prohibited for NO combinations. For OK combinations, set fixed-point as an input data format of cascade destination channels.

28.4.7 Trigger Control

28.4.7.1 Timer Trigger Input

The DFE uses 40 signals from timer D of the ATU4: compare-match A interrupts xy and compare-match B interrupts xy (x = 0, 2, 4, 6, 9; y = 0 to 3) as timer triggers.

Four compare-match A interrupt signals can be selected for each of three trigger flag functions (accumulation/decimation initialization flag, PH initialization flag, and filter initialization flag) with the PH initialization timer trigger select register (PITRG), accumulation/decimation initialization timer trigger select register (MITRG), and filter initialization timer trigger select register (FITRG). The selected signals are allocated to timer triggers 0 to 3 for each flag function.

Then, compare-match B interrupt signals with the same number as the compare-match A interrupt signals selected for the accumulation/decimation initialization flag are automatically allocated to the accumulation/decimation disable flag timer triggers 0 to 3.

Also, compare-match B interrupt signals with the same number as the compare-match A interrupt signals selected for the PH initialization flag are automatically allocated to the PH end flag timer triggers 0 to 3.

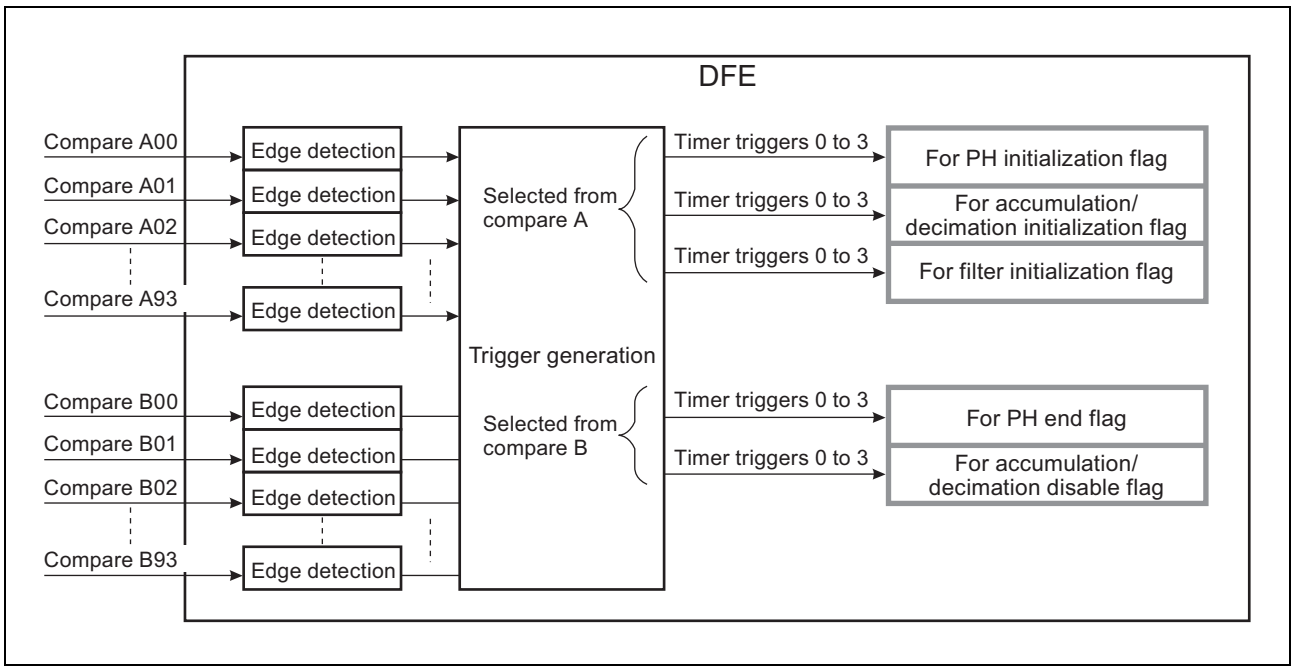


Figure 28.13 Timer Trigger Input

28.4.7.2 Trigger Flag Functions

When the channel enable bit (CTLACHn.EN) is 1, the DFE can generate flags of the following five functions for each channel by using software triggers, timer triggers 0 to 3 that are allocated to the trigger functions, and the trigger setting register (TRGCHn). (These flags are referred to as trigger flags hereafter.)

The accumulation/decimation processing is set with CTLBCHn.PRCSA, and the PH processing is set with CTLBCHn.PRCSB.

Whether the accumulation/decimation processing and PH processing are enabled or disabled can be checked by CTLBn.DISA and CTLBn.DISB, respectively. Before the DFE starts processing, in other words, when the channel is disabled (CTLACHn.EN = 0) and there is no data to be processed (STCHn.VALID = 0), CTLBn.DISA and CTLBn.DISB can be set to enable or disable processing.

For details, see **Section 28.2.3, CTLBCHn — Control Register B (n = 0 to 15)**.

Table 28.26 Trigger Flags

Flag Name	Description
Accumulation/decimation initialization flag	When the accumulation/decimation initialization flag is enabled by timer triggers or software triggers specified for each channel while accumulation/decimation processing is in progress or in the enabled or disabled state, the DFE initializes the counter of the accumulation circuit and performs accumulation/decimation processing for the subsequent input data.
Accumulation/decimation disable flag	When the accumulation/decimation disable flag is enabled by timer triggers or software triggers specified for each channel while accumulation/decimation processing is in progress or in the enabled state, the DFE disables accumulation/decimation processing for the subsequent input data. To enable accumulation/decimation processing again, enable the accumulation/decimation initialization flag.
PH initialization flag	When the PH initialization flag is enabled by timer triggers or software triggers specified for each channel while PH processing is in progress or in the enabled or disabled state, the DFE compares the next input data with the PH initialization value setting register value, not with a PH processing result as before, and after that the DFE compares the subsequent data with the PH processing result.
PH end flag	When the PH end flag is enabled by timer triggers or software triggers specified for each channel while PH processing is in progress or in the enabled state, the DFE disables PH processing for the subsequent input data. The PH end interrupt can be output when the PH end flag is enabled. To enable PH processing again, enable the PH initialization flag.
Filter initialization flag	When the filter initialization flag is enabled by timer triggers or software triggers specified for each channel, the DFE initializes the filter processing address pointer and performs FIR/IIR processing from the initial value.

Note 1. When timer triggers are used for the accumulation/decimation disable flag or PH end flag, timer triggers 0 to 3 are automatically allocated as described in **Section 28.4.7.1, Timer Trigger Input**.

Note 2. The PH initialization flag and PH end flag are only supported by channels 0 to 9.

Timer triggers and software triggers can be allocated to trigger flags for each channel. Timer triggers for accumulation/decimation disable flag and PH end flag are automatically allocated. Details of trigger flag operation are described in the sections of the accumulation circuit, PH circuit, and filtering circuit, respectively.

Table 28.27 Example of Trigger Flag Settings

	Accumulation/ Decimation Initialization Flag	Accumulation/ Decimation Disable Flag	PH Initialization Flag	PH Disable Flag	Filter Initialization Flag
Channel 0	Timer trigger 0 (compare A0)	Compare B0 is automatically allocated according to the timer trigger selection.	None	None	Timer trigger 3
Channel 1	None	None	Timer trigger 2 (compare A9)	Compare B9 is automatically allocated according to the timer trigger selection.	Software trigger
...	...	...	...	...	...
Channel 9	Timer trigger 3 (compare A5)	Compare B5 is automatically allocated according to the timer trigger selection.	None	None	None
Channel 10	Timer trigger 1	Software trigger	Not supported	Not supported	Timer trigger 0
...	...	...	...	...	...
Channel 15	None	None	Not supported	Not supported	None

Trigger flags are valid for data that is input after a software trigger or timer trigger is generated.

If timer trigger 0 is set for the accumulation initialization flag in channel 0, for example, accumulation processing is initialized for the first input data in channel 0 after a timer trigger is generated.

To process the same data in multiple channels, the trigger flag settings should be the same to keep consistency in processed data.

For example, when data input from the AD0 is specified for channels 0 and 1 and the accumulation initialization flag is used, the same accumulation initialization flag timer trigger should be set for both channels 0 and 1.

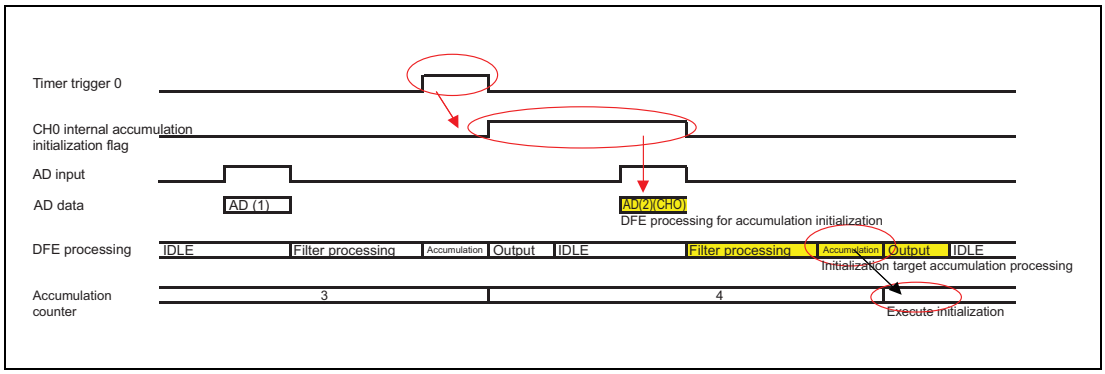


Figure 28.14 Example of Trigger Flag Operation

28.4.7.3 Trigger Flag Functions and Processing

Table 28.28 shows the relationship between the trigger flag functions and processing.

Table 28.28 Trigger Flag Functions and Processing

Register Setting		Trigger Flag Function		Input Data Processing after Trigger Flag is Enabled			
		Accumulation/ decimation	PH	Output data write	Output data interrupt	Condition match interrupt	PH update notification
Accumulation/ decimation disabled	PH/comparison disabled	Flag is invalid	Flag is invalid	Provided	Not provided	Not provided	Not provided
	PH performed	Flag is invalid	Enabled	Provided	Not provided	Not provided	Provided
			Disabled	Provided	Not provided	Not provided*	Not provided
	Comparison performed	Flag is invalid	Flag is invalid	Provided	Not provided	Provided	Not provided
Accumulation/ decimation performed	PH/comparison disabled	Enabled	Flag is invalid	Provided	Provided	Not provided	Not provided
		Disabled	Flag is invalid	Not provided	Not provided	Not provided	Not provided
	PH performed	Enabled	Enabled	Provided	Provided	Not provided	Provided
			Disabled	Provided	Provided	Not provided*1	Not provided
		Disabled	Flag is invalid	Not provided	Not provided	Not provided	Not provided
	Comparison performed	Enabled	Flag is invalid	Provided	Provided	Provided	Not provided
		Disabled	Flag is invalid	Not provided	Not provided	Not provided	Not provided

Note 1. The interrupt is output when the PH end flag is enabled.

When accumulation/decimation processing and PH/comparison processing are used in the same channel with the accumulation/decimation count specified, PH processing is performed on each specified count.

If accumulation/decimation processing is disabled by the trigger flag function, PH/comparison processing is not performed, either. Do not input software triggers or timer triggers for the trigger flag function for PH initialization and PH end trigger flag functions to a channel with accumulation/ decimation processing disabled.

28.4.8 Input Data Control

The DFE allows AD input (from Delta-Sigma AD0 to AD7 and ADC0/ADC1), cascade input (to perform filter calculation of the DFE's filter calculation result again), and software input (by the CPU or DMA).

28.4.8.1 AD Input

The DFE compares the tag value (channel tag hereafter) set in the TAG bits in CTLACHn for each channel and the AD tag input from the AD, and performs processing in the channel that has the matched tag value. This function allows single piece of AD data to be input to processing in multiple channels.

28.4.8.2 Cascade Input

The DFE has a cascade input function to input filter processing result again. This cascade input function performs rounding processing for 32-bit accumulation circuit output data to 16-bit data, and then performs filter processing again. In integer mode, when the accumulation circuit output data is a positive value and the 16 high-order bits contain a valid value, the output data is rounded to a 16-bit positive maximum value ($7FFF_H$). When the accumulation circuit output data is a negative value and the 16 high-order bits contain a valid value, the output data is rounded to a 16-bit negative minimum value (8000_H). If rounding of both positive and negative values occurs, the cascade rounding error flag is set to 1. In fixed-point mode, the accumulation circuit output data is added to 0000 8000_H and rounding processing is performed. As a result of this addition, if an overflow occurs in 32-bit positive value, the added data is rounded to a positive maximum value ($7FFF_H$) and the cascade rounding error flag is set.

The CAEN bits in the control register CTLACHn can control enabling or disabling the cascade input function and enabling or disabling the output data register mask function.

Table 28.29 CAEN Flag Function

CAEN Bits	Description
00_B	Cascade input is disabled. Calculation result is written to the output data register. (Initial value)
01_B	Setting prohibited
10_B	Cascade input is enabled. Calculation result is written to the output data register.
11_B	Cascade input is enabled. Calculation result is not written to the output data register.

When cascade input is enabled, the cascade input destination channel is specified by using the tag value for cascade (cascade tag hereafter). The cascade tag is set by the CATAG bits in CTLACHn. The DFE compares the cascade tag with the channel tag, and performs cascade input filter processing in the channel where the tags matched. The following figures show operations when cascade input is enabled.

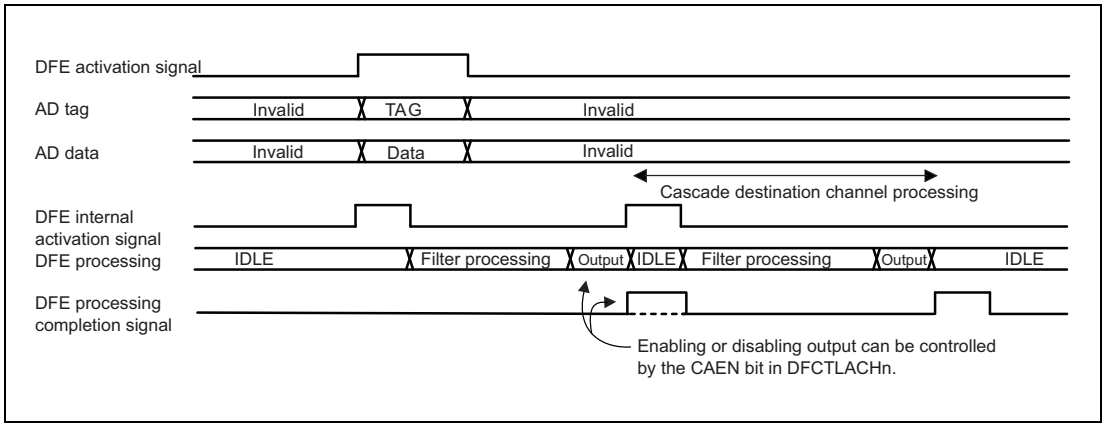


Figure 28.15 Cascade Input Processing (Normal Processing)

When cascade input is enabled and decimation is specified, a decimation result becomes cascade input data. When decimation count is set to 4, for example, the data after filter processing of the fourth AD input becomes cascade input data.

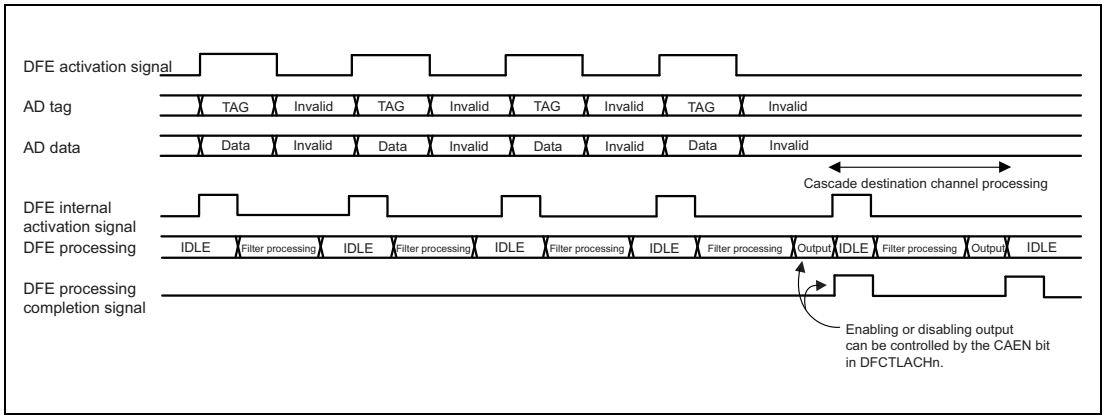


Figure 28.16 Cascade Input Processing (Decimation: 4 Times)

28.4.8.3 Software Input

The DFE has a function to input DFE processing target data through the peripheral bus by using the CPU or DMA. (This function is referred to as software input function hereafter.) DFE processing can be started by writing filter processing target data and a tag value to the software input data register (DI) from the peripheral bus.

28.4.8.4 Dispatch

A channel to be processed is selected by activation signals of AD input (Delta-Sigma AD and ADC), cascade input, and software input and the channel tag, and then processing target data is written to the input data register of each channel. This processing is called dispatch.

After processing target data is written to the input data register, the input data register is enabled until the accumulation circuit processing is completed. If new data is written for the same channel while the input data register is enabled, the data is overwritten causing an input data overwrite error (DIOW).

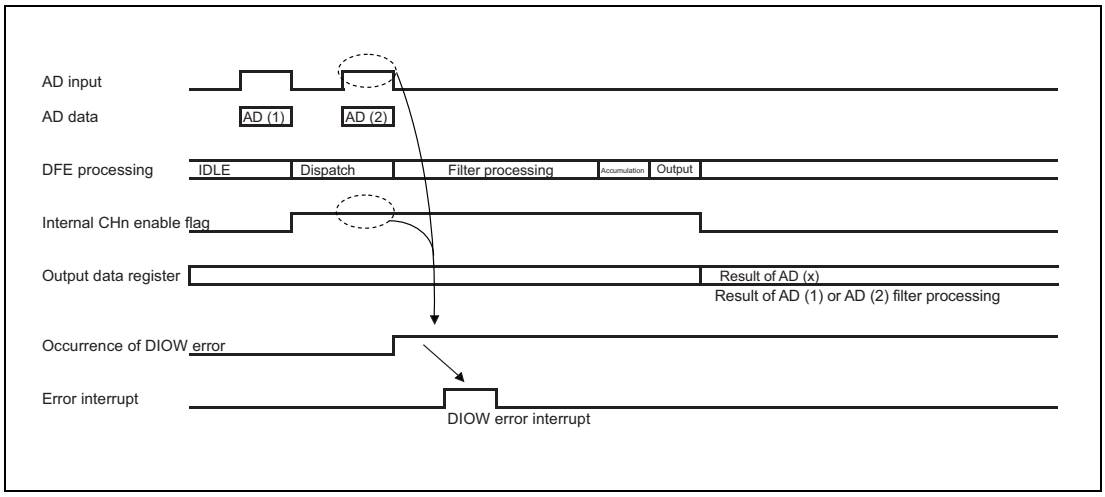


Figure 28.17 Input Data Overwrite (DIOW) Error

When the input data register of each channel is enabled, filter processing target data is arbitrated. The highest priority is given to the processing for channel 0, the second highest priority is given to the processing for channel 1, and the lowest priority is given to the processing for channel 15. The input data register value of arbitrated channels is input to the filtering circuit where FIR/IIR processing is applied to the input data register value according to the control register value.

If the same data is set for multiple channels, processing is also performed for all of the specified channels in the channel priority order. If the trigger flag function is enabled, it is not applied until the next input data processing.

For example, if the same input data is set for multiple channels and the accumulation processing is set and the same accumulation/decimation disable trigger is set for the channels, the accumulation is not disabled until the next input data processing if any unprocessed channel exists when the disable trigger is input.

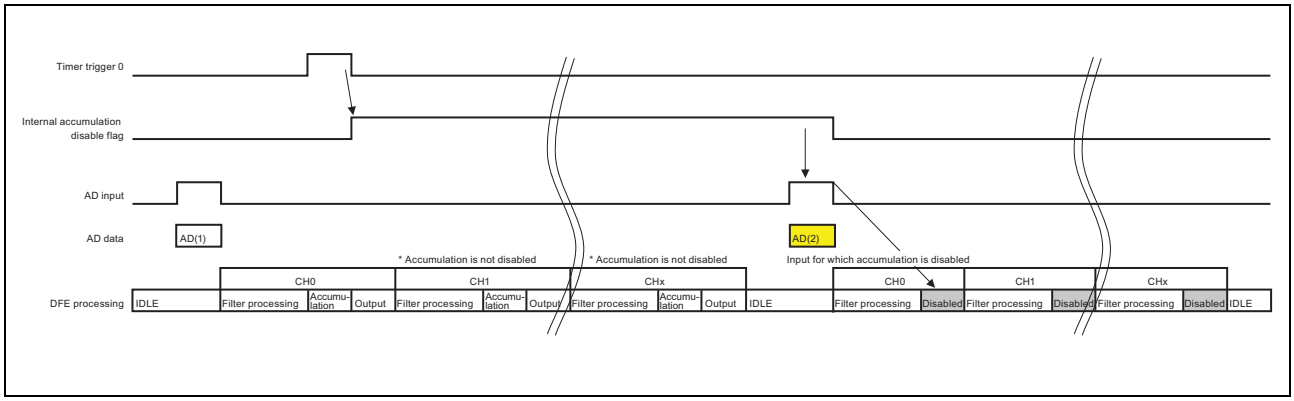


Figure 28.18 Processing when the Same Input Data is Set for Multiple Channels

28.4.9 Filtering Circuit

The filtering circuit accepts a request from channel 0 to channel 15 selected in the dispatch circuit, and starts FIR or IIR filter processing. Another request cannot be accepted during execution of filter processing. Upon completion of processing in the accumulation circuit, the DFE accepts a new request. Coefficients required for filtering are stored in the coefficient memory and input data to be used for FIR processing and intermediate data to be used for IIR processing are stored in the data memory. These memory areas are allocated for each channel.

28.4.9.1 FIR

Figure 28.19 shows the FIR filter configuration.

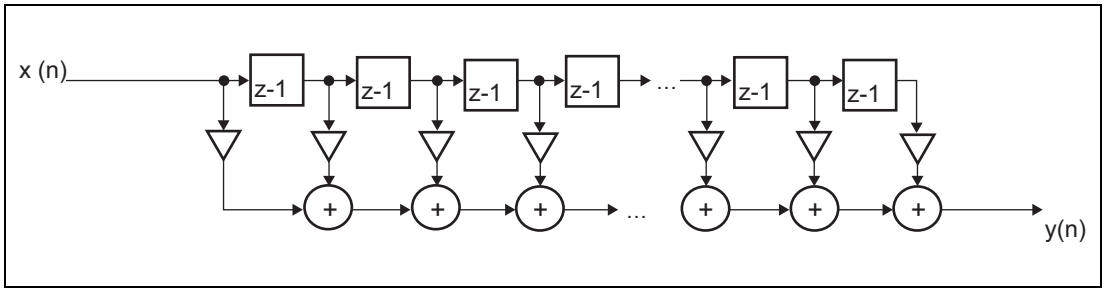


Figure 28.19 FIR Filter Configuration

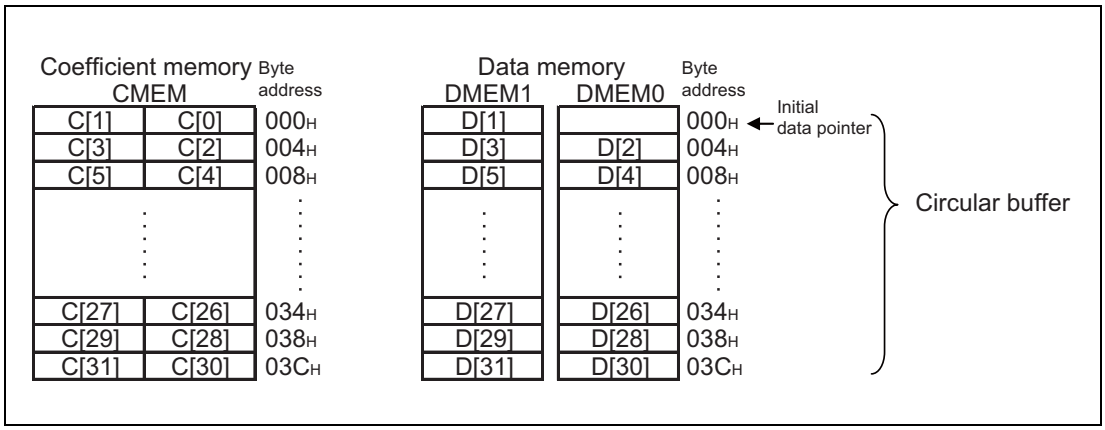


Figure 28.20 Memory Configuration for FIR Filter (32TAP Selected)

Before starting FIR filter processing, set values for the coefficient memory and data memory buffer areas. For 8-tap FIR, set coefficient values to be used for coefficient memory C[0] to C[7] and set the circular buffer area in the data memory to 0000_H.

While the filter processing initialization flag is enabled, the DFE writes 0 to circular buffers of the number of taps, and then starts the FIR processing. In the case of 8-tap FIR, for example, the DFE writes 0 to circular buffers of D[0] to D[7] and then starts FIR processing.

28.4.9.2 IIR (Single Stage)

Figure 28.21 shows the IIR filter configuration (single stage).

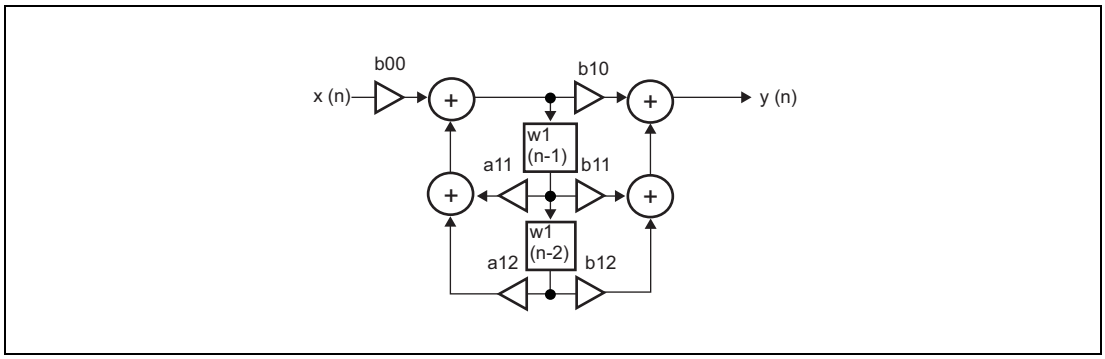


Figure 28.21 IIR Filter Configuration (Single Stage)

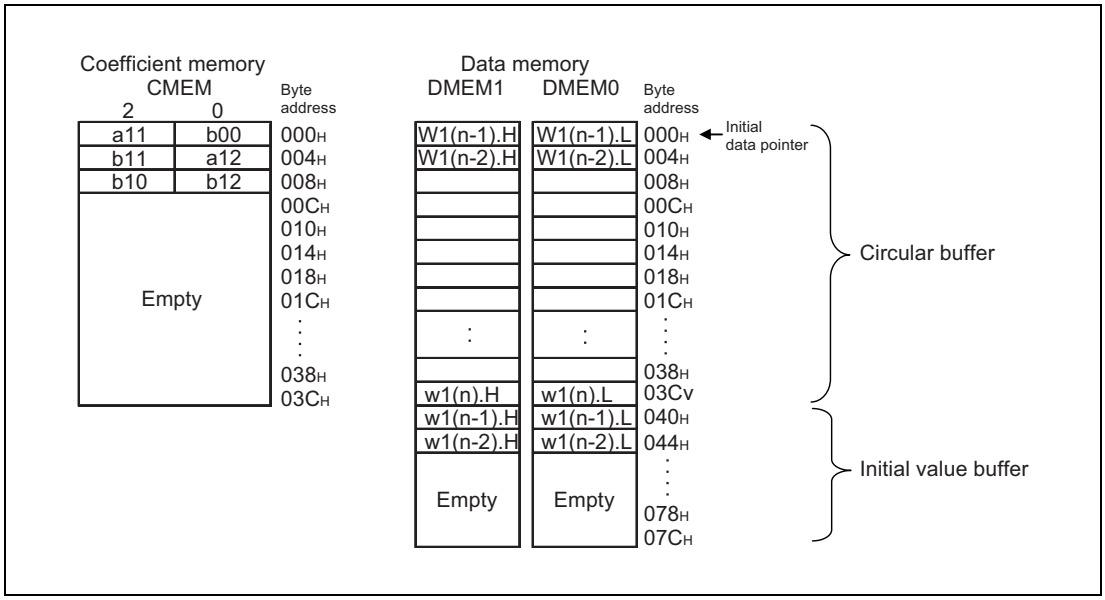


Figure 28.22 Memory Configuration for IIR Filter (Single Stage)

Before starting IIR filter processing, set values for the coefficient memory and data memory buffer areas. Set the intermediate value data for the circular buffer area and the initial value buffer area in the data memory.

While the filter processing initialization flag is enabled, the DFE initializes the data pointer of the circular buffer to 0, writes the intermediate value data stored in the initial value buffer to the relevant circular buffer, and then performs IIR processing.

28.4.9.3 IIR (Two Stages)

Figure 28.23 shows the IIR filter configuration (two stages).

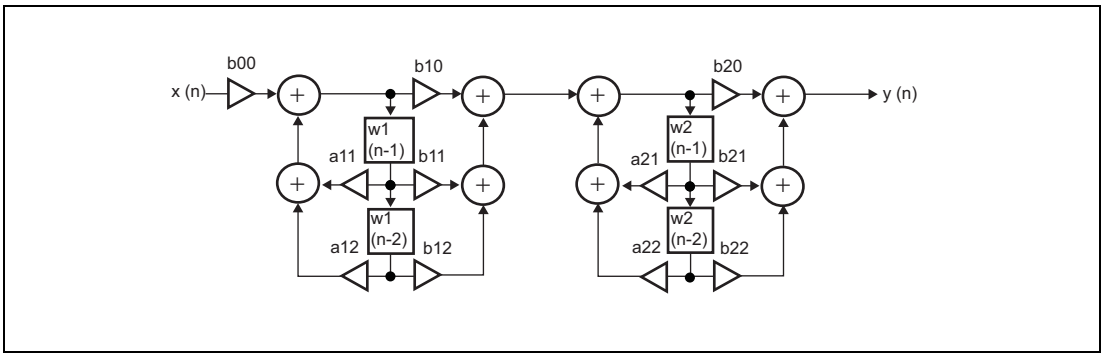


Figure 28.23 IIR Filter Configuration (Two Stages)

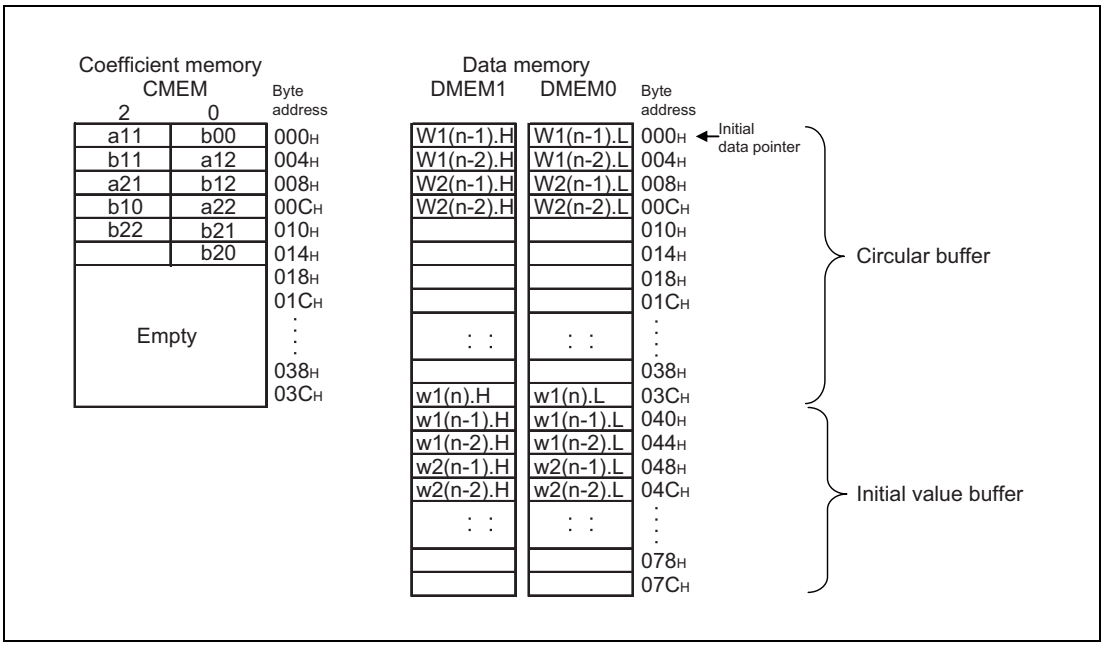


Figure 28.24 Memory Configuration for IIR Filter (Two Stages)

Before starting IIR filter processing, set values for the coefficient memory and data memory buffer areas. Set the intermediate value data for the circular buffer area and the initial value buffer area in the data memory.

While the filter processing initialization flag is enabled, the DFE initializes the data pointer of the circular buffer to 0, writes the intermediate value data stored in the initial value buffer to the relevant circular buffer, and then performs IIR processing.

28.4.9.4 IIR (Three Stages)

Figure 28.25 shows the IIR filter configuration (three stages).

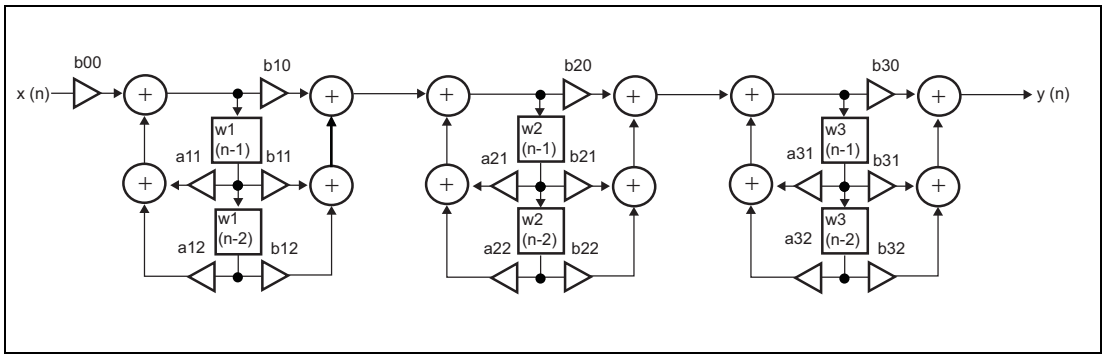


Figure 28.25 IIR Filter Configuration (Three Stages)

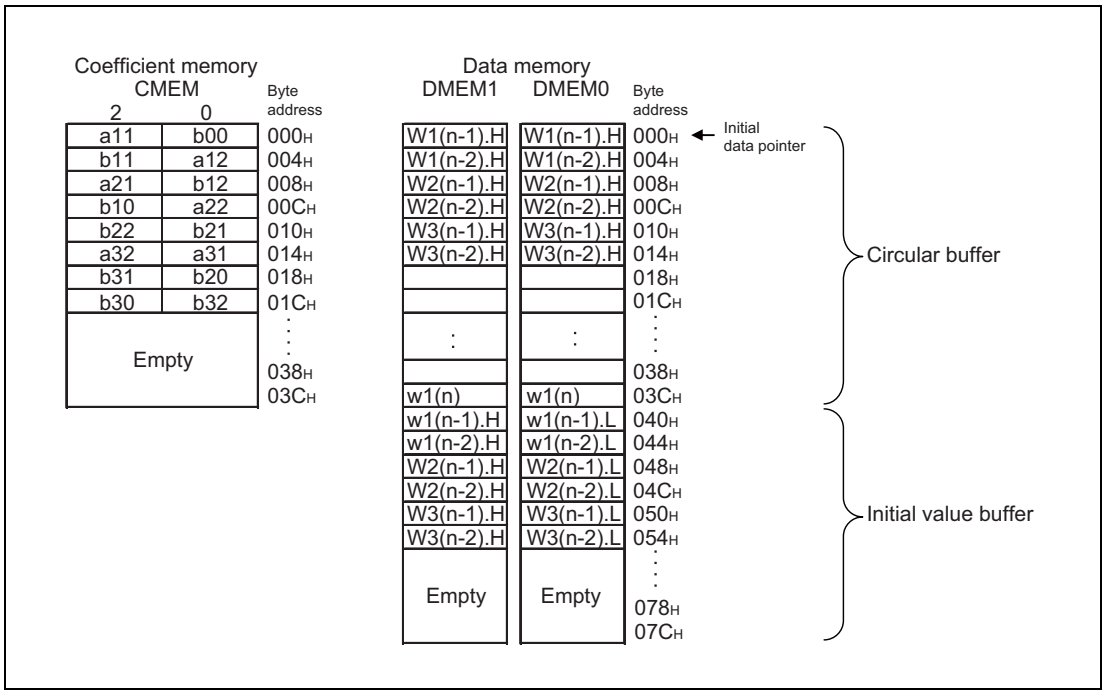


Figure 28.26 Memory Configuration for IIR Filter (Three Stages)

Before starting IIR filter processing, set values for the coefficient memory and data memory buffer areas. Set the intermediate value data for the circular buffer area and the initial value buffer area in the data memory.

While the filter processing initialization flag is enabled, the DFE initializes the data pointer of the circular buffer to 0, writes the intermediate value data stored in the initial value buffer to the relevant circular buffer, and then performs IIR processing.

28.4.9.5 Errors in Filtering Circuit

If multiplication of 8000_H*8000_H is performed by the multiplier, a multiplication error occurs and the multiplication error flag is set to 1.

To prevent an overflow in the multiplication result addition processing, six guard bits are provided. To round data with guard bits to 32 bits, rounding operation is performed. When guard bits contain a valid value after the rounding operation, the error flag is set to 1. This error is called guard bit error.

When the calculation result is larger than the 32-bit positive maximum value ($7FFF\_FFFF_H$), saturation processing is performed for $7FFF\_FFFF_H$. When the calculation result is smaller than the 32-bit negative minimum value (8000_0000_H), saturation processing is performed for 8000_0000_H.

28.4.10 Accumulation Circuit

The accumulation circuit has a function to perform absolute value conversion, accumulation processing, or decimation processing for the filter processing result data that is output from the filtering circuit and to output the processing result data to the output circuit and the PH circuit. When the cascade function is used, the accumulation circuit sends the accumulation circuit output data to the dispatch circuit.

28.4.10.1 Absolute Value Calculation

The filter processing result data that is output from the filtering circuit is converted to an absolute value. When the ABS bit in CTLBCHn is 1, absolute value conversion is performed. When the ABS bit is 0, absolute value conversion is not performed. If the filter processing result data that is output from the filtering circuit is 8000_0000_H, an absolute value error is generated. The absolute value conversion result at this time is $7FFF\_FFFF_H$.

28.4.10.2 Summary of Accumulation Processing

In the accumulation processing, the filter processing result data that is input to the accumulation circuit is right shifted (9 bits) arithmetically, and then this filter processing result data is added to the filter processing result data that has been input for each channel.

The data format after the accumulation processing is (10.22). Accumulation processing can be performed up to 511 times. After processing for the specified accumulation processing count has been completed, accumulation processing is performed again.

When the PRCSA bits in CTLBCHn are 01_B, accumulation processing is performed. Specify the accumulation processing count by the SELA bits in CTLBCHn by selecting one of the four accumulation/decimation count setting registers (ACA to ACD).

28.4.10.3 Operation of Accumulation Processing

Operation of accumulation processing is explained using an example when ACA = 000_H, ACB = 001_H/ ACC = 002_H, and CTLBCHn.PRCSA = 01_B.

When the SELA bits in CTLBCHn are 00_B, ACA is selected and accumulation processing is performed with an accumulation count set value of 0.

The 9-bit arithmetic right-shift result data (DADD (1)) from the filter processing result data (D (1)) is written to the output data register and an output data interrupt request is issued.

In accumulation processing with an accumulation count set value of 0, one-time filter processing result is right shifted (9 bits) arithmetically without addition processing and an interrupt request is issued. This processing is repeated subsequently.

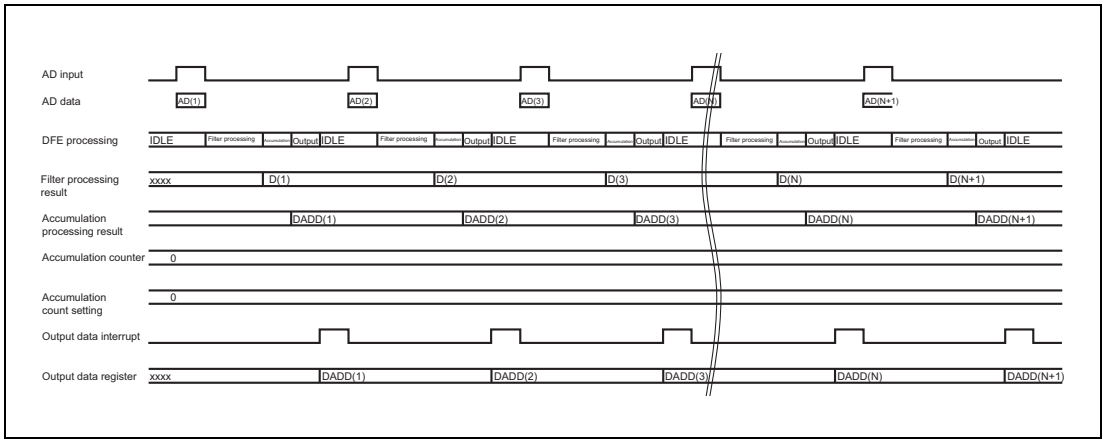


Figure 28.27 Accumulation Processing (Set Count Value = 0)

When the SELA bits in CTLBCHn are 01_B, the ACB register is selected and accumulation processing is performed with a set accumulation count value of 1.

When the filter processing result data (D (1)) is input to the accumulation circuit, the 9-bit arithmetic right-shift result data (DADD (1)) from the filter processing result data (D (1)) is written to the accumulator. After that, when the filter processing result data (D (2)) is input to the accumulation circuit, the filter processing result data (D (2)) is right shifted (9 bits) arithmetically (DADD (2)) and this data is added to the accumulator value (DADD (1)). The addition result (DADD (1) + DADD (2)) is written to the output data register. After this writing has been completed, an output data interrupt request is issued.

When the accumulation count is set to 1, 9-bit arithmetic right shift is made for two filter processing results and these results are added, and then an interrupt request is issued once. This processing is repeated subsequently.

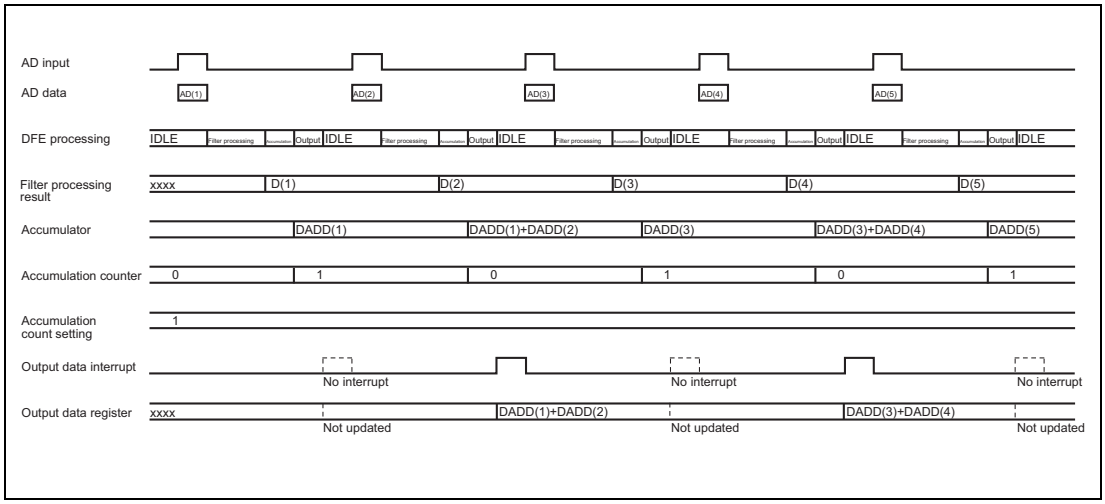


Figure 28.28 Accumulation Processing (Set Count Value = 1)

When the SELA bits in CTLBCHn are 10_B, the ACC register is selected and accumulation processing is performed with a set accumulation count value of 2.

When the filter processing result data (D (1)) is input to the accumulation circuit, the 9-bit arithmetic right-shift result data (DADD (1)) from the filter processing result data (D (1)) is written to the accumulator. After that, when the filter processing result data (D (2)) is input to the accumulation

circuit, the filter processing result data (D (2)) is right shifted (9 bits) arithmetically (DADD (2)) and this data is added to the accumulator value (DADD (1)). The addition result (DADD (1) + DADD (2)) is written to the accumulator again. Next, when the filter processing result data (D (3)) is input to the accumulation circuit, the filter processing result data (D (3)) is right shifted (9 bits) arithmetically (DADD (3)) and this data is added to the accumulator value (DADD (1) + DADD (2)). The addition result (DADD (1) + DADD (2) + DADD (3)) is written to the output data register. After this writing has been completed, an output data interrupt request is issued.

When the accumulation count is set to 2, 9-bit arithmetic right shift is made for three filter processing results and these results are added, and then an interrupt request is issued once. This processing is repeated subsequently.

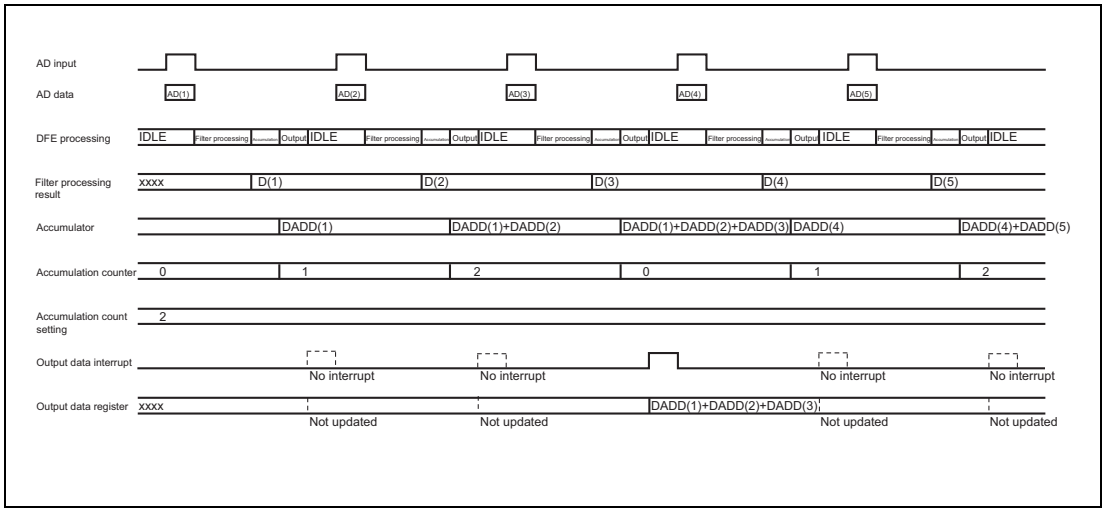


Figure 28.29 Accumulation Processing (Set Count Value = 2)

28.4.10.4 Operation of Accumulation Processing (with Trigger Flag)

The following describes operation of accumulation processing when the accumulation initialization flag and the accumulation disable flag are enabled.

When the accumulation initialization flag is enabled by a timer trigger or a software trigger while accumulation is specified (PRCSA = 01_B in CTLBCHn), the accumulation counter is initialized and accumulation processing is performed as the first data of the accumulation processing in the DFE‘s accumulation processing of the immediately following target channel.

In case the accumulation initialization flag is enabled and the accumulation count is set to 2, for example, when data (D (3)) is input to the accumulation circuit third time, the 9-bit arithmetic right-shift (DADD (3)) from the third data (D (3)) is written to the accumulator. The accumulation result value (DADD (1) + DADD (2)) is overwritten.

The data (D (4)) that is input next to the accumulation circuit is added to the data (D (5)) that is input after that to the accumulation circuit. Three accumulation results ((DADD (3) + DADD (4) + DADD (5)) are written to the output data register after the accumulation initialization flag is enabled, and an output data interrupt request is issued.

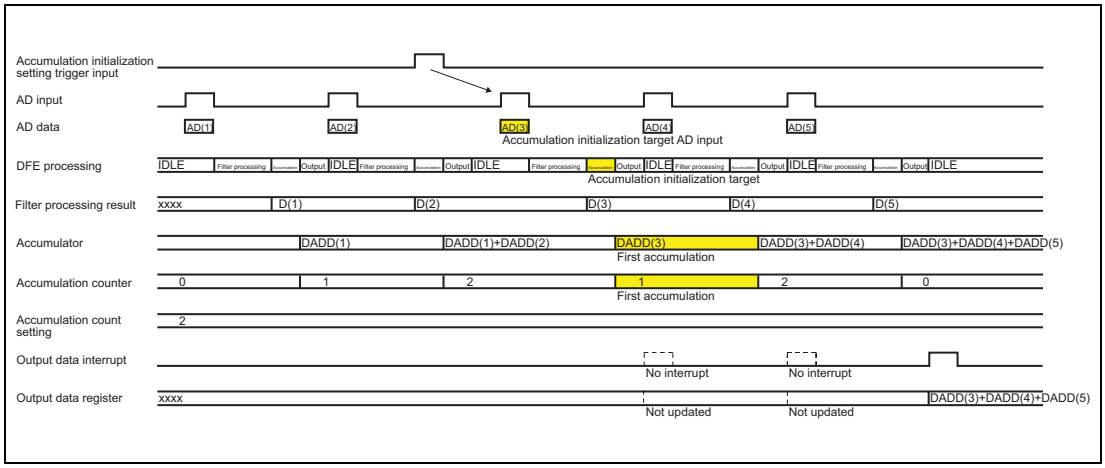


Figure 28.30 Accumulation Processing (with the Initialization Flag Enabled)

When the accumulation disable flag is enabled by a timer trigger or a software trigger while accumulation is specified (PRCSA = 01_B in CTLBCHn), the DFE's accumulation processing of the immediately following target channel is not performed and the internal accumulation disable flag is asserted. No accumulation is performed in the subsequent processing.

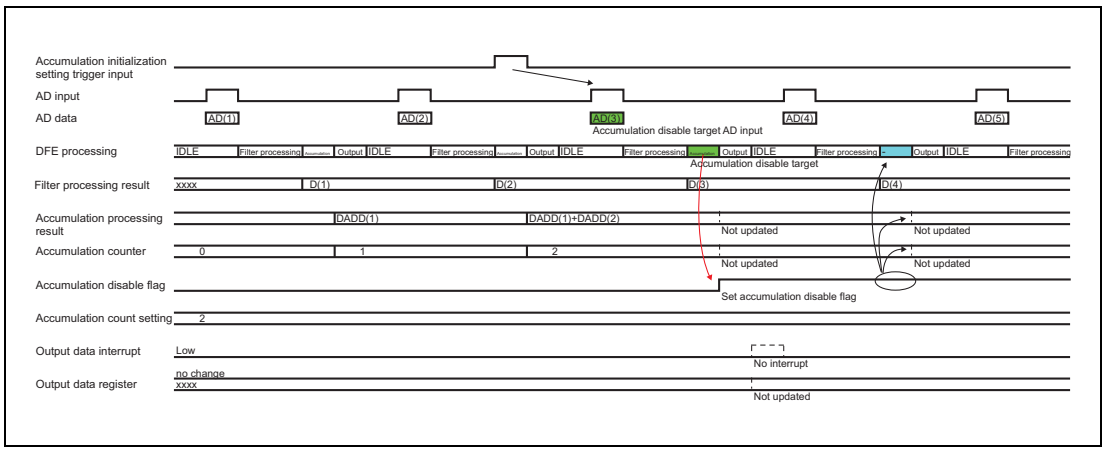


Figure 28.31 Accumulation Processing (with the Accumulation Disable Flag Enabled)

To perform accumulation processing again while the accumulation disable flag is asserted, enable the accumulation initialization flag.

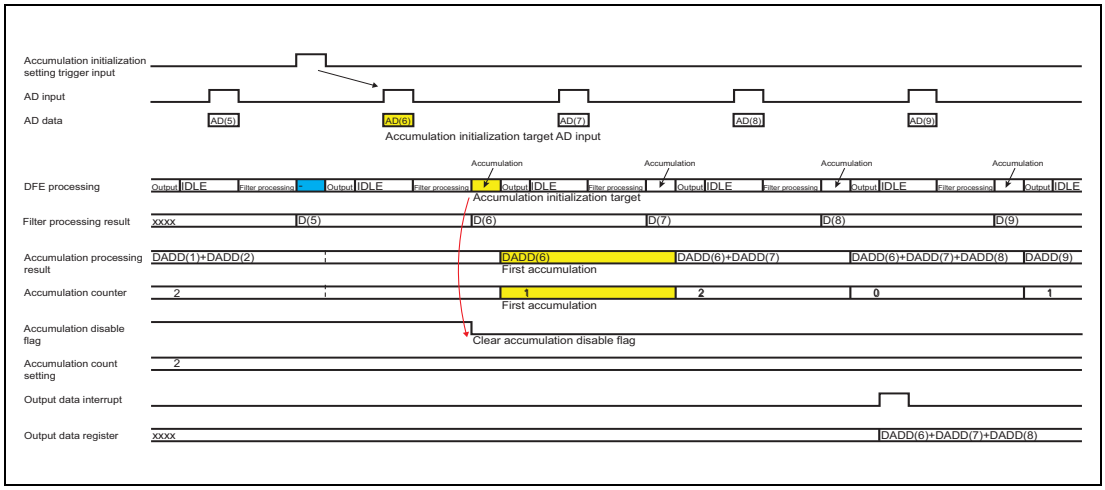


Figure 28.32 Accumulation Processing (Disable to Enable)

28.4.10.5 Summary of Decimation Processing

In the decimation processing, filter processing result data that is input to the accumulation circuit is decimated. The decimation processing count can be set to up to 511. (Data is output once every 512 times.) After processing for the specified decimation count has been completed, decimation processing is performed again.

When the PRCSA bits in CTLBCHn are 10_B, decimation processing is performed. Specify the decimation processing count by the SELA bits in CTLBCHn by selecting one of the four accumulation/decimation count setting registers (ACA to ACD).

28.4.10.6 Operation of Decimation Processing

Operation of decimation processing is explained using an example when ACA = 0_H, ACB = 1_H, ACC = 2_H, and PRCSA bits in CTLBCHn = 10_B.

When the SELA bits in CTLBCHn are 00_B, ACA is selected and decimation processing is performed with a decimation count set value of 0.

In decimation processing with a decimation count set value of 0, one-time filter processing result is output as it is and an interrupt request is issued. This processing is repeated subsequently.

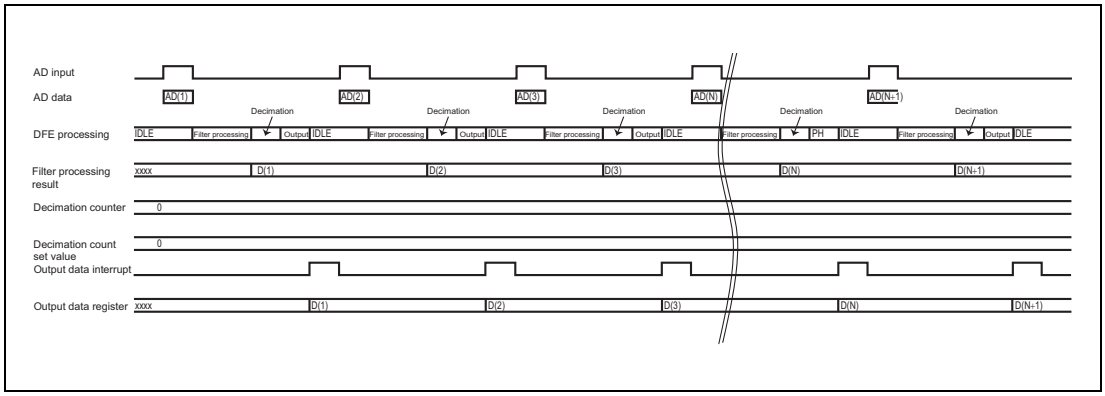


Figure 28.33 Decimation Processing (Set Count Value = 0)

When the SELA bits in CTLBCHn are 01_B, the ACB register is selected and decimation processing is performed with a set decimation count value of 1.

When the filter processing result data (D (1)) is input to the accumulation circuit, the decimation counter is incremented. At this time, the accumulation circuit does not output the input filter processing result data (D (1)) to the output circuit because the decimation counter value differs from the set decimation count value.

After that, when the filter processing result data (D (2)) is input to the accumulation circuit, the accumulation circuit outputs the input filter processing result data (D (2)) to the output circuit because the decimation counter value is equal to the set decimation count value. The output circuit writes the data (D (2)) that is input from the accumulation circuit to the output data register and issues an output data interrupt request.

When the decimation count is set to 1, an interrupt request is issued once in two filter processings. This processing is repeated subsequently.

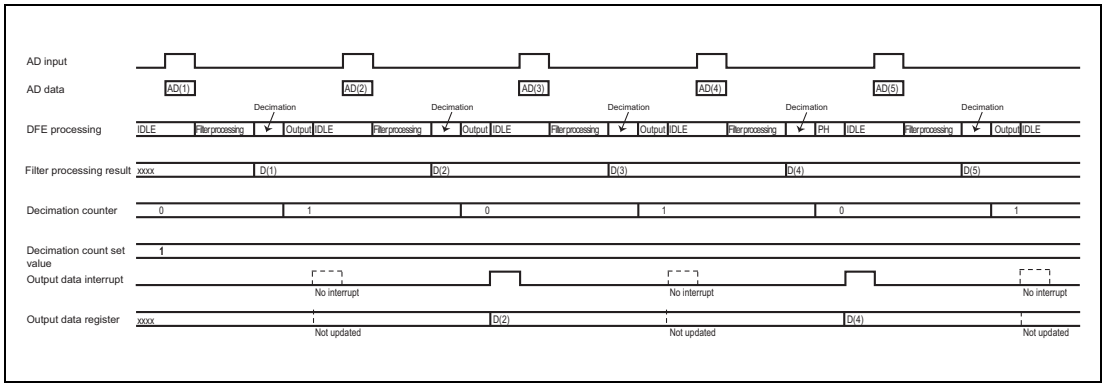


Figure 28.34 Decimation Processing (Set Count Value = 1)

When the SELA bits in CTLBCHn are 10_B, the ACC register is selected and decimation processing is performed with a set decimation count value of 2.

When the filter processing result data (D (1)) is input to the accumulation circuit, the decimation counter is incremented. At this time, the accumulation circuit does not output the input filter processing result data (D (1)) to the output circuit because the decimation counter value differs from the set decimation count value.

Then, when the filter processing result data (D (2)) is input to the accumulation circuit, the decimation counter is incremented. At this time, the accumulation circuit does not output the input filter processing result data (D (2)) to the output circuit because the decimation counter value differs from the set decimation count value.

After that, when the filter processing result data (D (3)) is input to the accumulation circuit, the accumulation circuit outputs the input filter processing result data (D (3)) to the output circuit because the decimation counter value is equal to the set decimation count value. The output circuit writes the data (D (3)) that is input from the accumulation circuit to the output data register and issues an output data interrupt request.

When the decimation count is set to 2, an interrupt request is issued once in three filter processings. This processing is repeated subsequently.

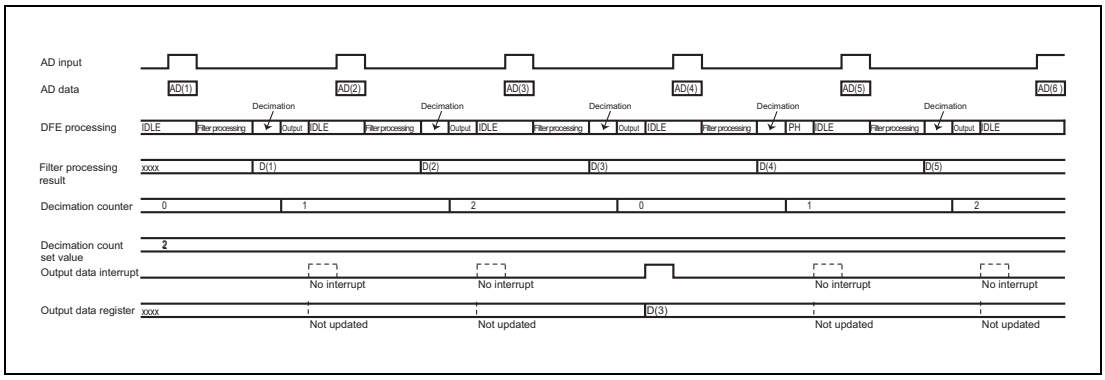


Figure 28.35 Decimation Processing (Set Count Value = 2)

28.4.10.7 Operation of Decimation Processing (with Trigger Flag)

The following describes operation of decimation processing when the decimation initialization flag and the decimation disable flag are enabled. When the decimation initialization flag is enabled by a timer trigger or a software trigger while decimation is specified (PRCSA = 10_B in CTLBCHn), the decimation counter is initialized and decimation processing is performed as the first data of the decimation processing in the DFE's decimation processing of the immediately following target channel.

In case the decimation initialization flag is enabled and the decimation count is set to 2, for example, when data (D (3)) is input to the accumulation circuit third time, the third data (D (3)) is not output from the accumulation circuit, but is processed as the first input data. Subsequently, the accumulation circuit does not output data (D (4)) that is input next to the accumulation circuit as usual, but outputs data (D (5)) that is input after that to the output circuit.

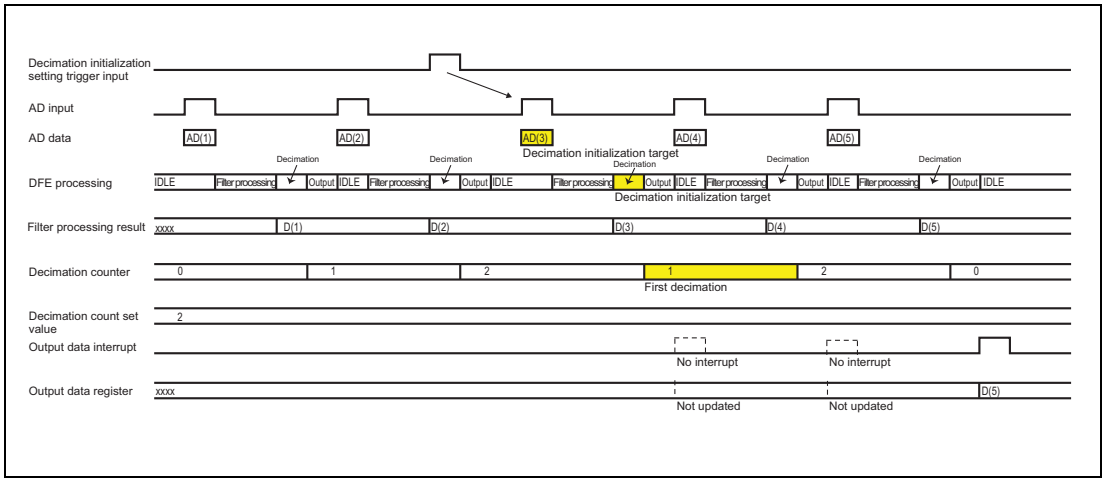


Figure 28.36 Decimation Processing (with the Initialization Flag Enabled)

When decimation is specified (PRCSA = 10_B in CTLBCHn) and the decimation disable flag is enabled, the internal decimation disable flag is asserted. In the subsequent processing, the decimation counter is disabled. While decimation is disabled, no output data interrupt request is issued.

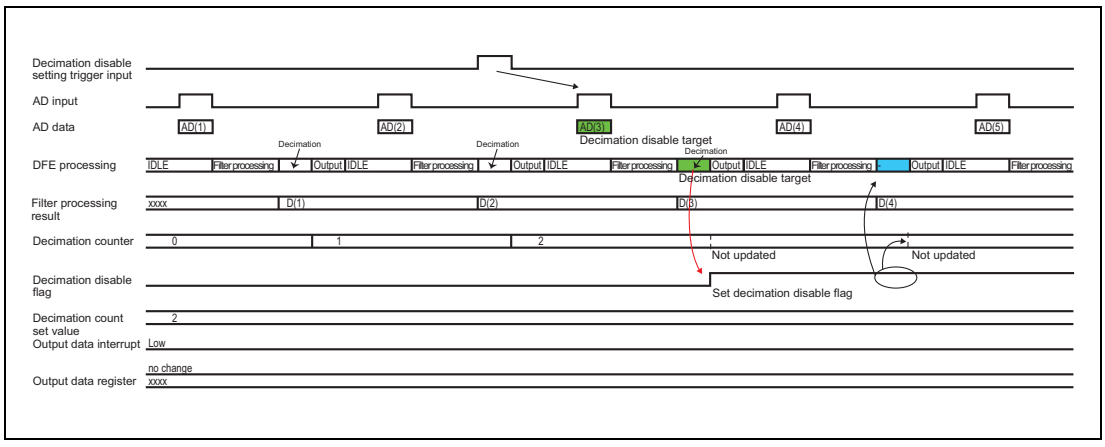


Figure 28.37 Decimation Processing (with the Decimation Disable Flag Enabled)

To perform decimation processing again while the internal decimation disable flag is asserted, enable the decimation initialization flag.

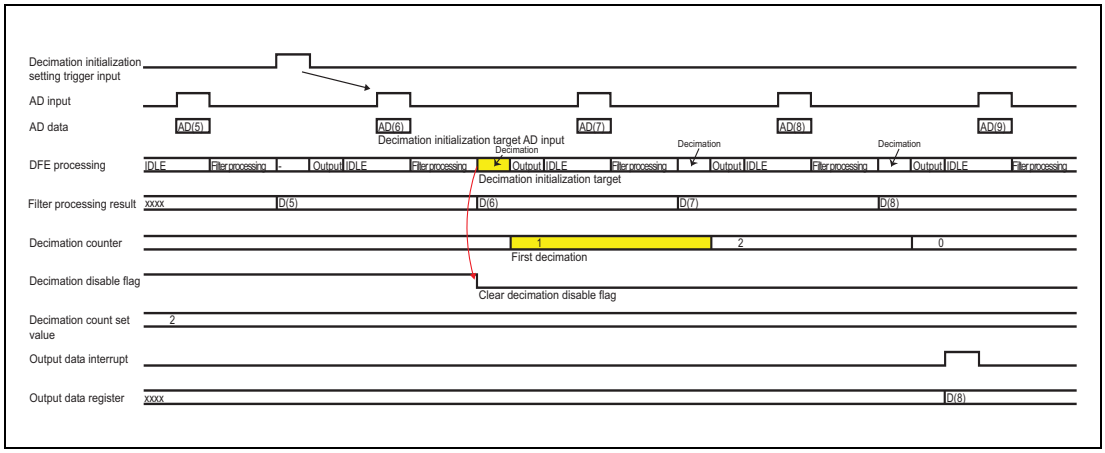


Figure 28.38 Decimation Processing (Disable to Enable)

28.4.11 PH Circuit

The PH circuit performs PH processing or comparison processing for data that is output from the accumulation circuit. When the accumulation/decimation count is specified, PH processing is performed on each specified count.

In PH processing, PH results are written to the PH result register. A condition match interrupt is generated at the end of PH processing. In comparison processing, accumulation circuit output data is compared with the value of the comparison target register. When the comparison result is true, a condition match interrupt is generated.

28.4.11.1 Summary of PH Processing

In PH processing, data that is input to the PH circuit is compared with the data that has been input to the PH circuit for each channel, and the larger value is written to the PH result register as a PH result. The PH initial value can be selected by using the SELB2 bits in CTLBCHn. Once PH processing starts, it is performed continuously until the PH processing end flag is enabled. After the PH processing flag is enabled, PH processing of data that is input to the PH circuit is not restarted until the PH initialization flag is enabled.

28.4.11.2 Operation of PH processing

Operation of PH processing is explained using an example when PRCSB = 01_B (PH specified) in CTLBCHn.

For the first input data to the target channel, PH processing is performed using the PH initial value register value and PH circuit input data value. After that, PH processing is performed using the PH result register value and PH circuit input data value.

During PH processing, when the PH initialization flag is enabled, the next input data for the target channel is compared with the PH initial value register value, not with the PH result register value.

When the PH end flag is enabled, PH processing is terminated.

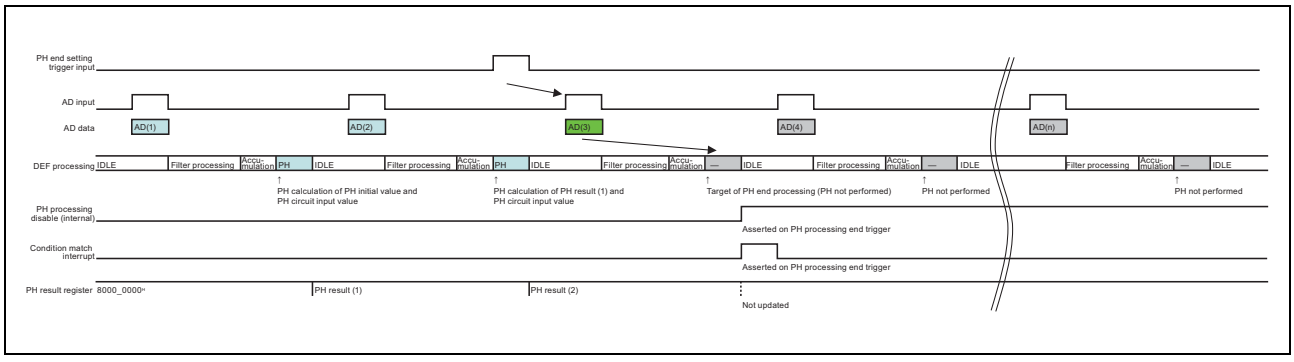


Figure 28.39 PH Processing (from PH Processing Start to PH End Flag)

NOTE

n = 0 to 9

To restart PH processing, enable the PH initialization flag.

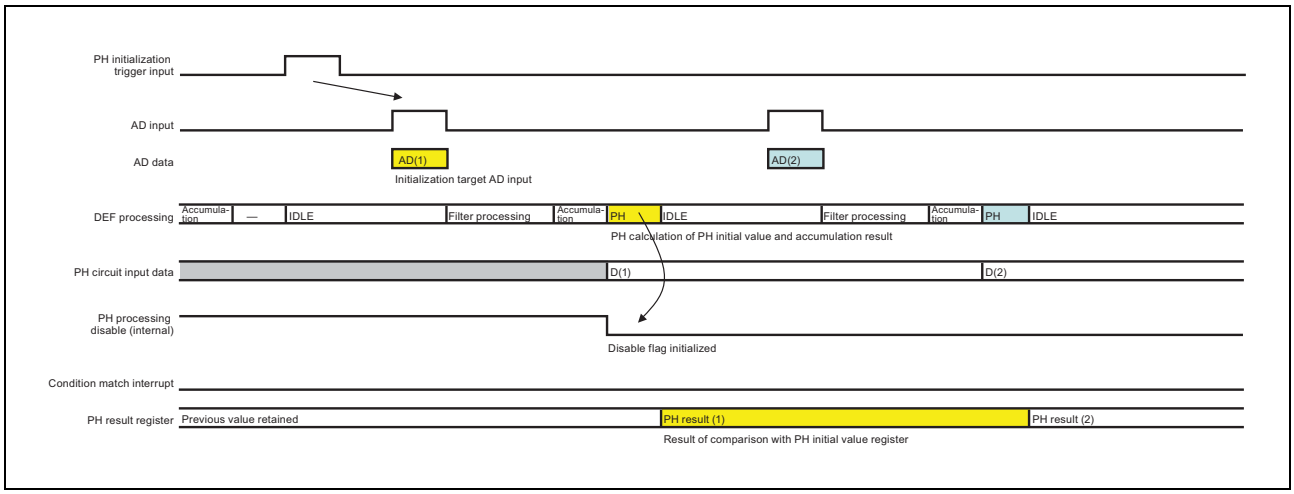


Figure 28.40 PH Processing (PH Initialization Enabled during PH Processing is Disabled)

28.4.11.3 Comparison

In comparison processing, the PH circuit input data is compared with the value of comparison target registers (CPA to CPD) for each channel. When the comparison result is true, the CND flag (condition match flag) is asserted and a condition match interrupt request is issued. Comparison processing is not affected by trigger flags. When the PRCSB bits in CTRLCHn are 10_B, comparison processing is performed.

The following comparison calculations can be performed.

Comparison	Description
Equal to (= =)	“Equal to (= =)” calculation is performed between the PH circuit input data and the selected value to be compared. When the comparison result is “equal to,” it is true.
Equal to or less than (≤)	“Equal to or less than (≤)” calculation is performed between the PH circuit input data and the selected value to be compared. When the comparison result is “equal to or less than,” it is true.
Equal to or more than (≥)	“Equal to or more than (≥)” calculation is performed between the PH circuit input data and the selected value to be compared. When the comparison result is “equal to or more than,” it is true.
Less than (<)	“Less than (<)” calculation is performed between the PH circuit input data and the selected value to be compared. When the comparison result is “less than,” it is true.
More than (>)	“More than (>)” calculation is performed between the PH circuit input data and the selected value to be compared. When the comparison result is “more than,” it is true.

Once the comparison calculation result is true and the condition match bit (CND in STCHn) is set to 1, the condition match bit remains 1 even if the subsequent comparison calculation result is false. Use the clear status register to clear the condition match bit to 0.

NOTE

n = 0 to 9

28.4.12 Output Circuit

28.4.12.1 Floating-Point Conversion Circuit

This circuit performs IEEE754 floating-point conversion of (1.31) format data that is input to the output circuit and (10.22) format data after accumulation. Specify floating-point conversion processing by setting the PRCSC bit in the control register B (CTLBCHn) to 1_B.

28.4.12.2 Output Register Circuit

The output register circuit has 32-bit registers for 16 channels. Input calculation result data or floating-point conversion result is written to the output register for each channel.

When a result of calculation with CTLBCHn.PRCSA = 01_B (accumulation) or CTLBCHn.PRCSA = 10_B (decimation) is written to an output register, the DOEN bit in STCHn is set and an interrupt request is generated. When filter processing is performed with CTLBCHn.PRCSA = 00_B, the filtering result is stored in an output register but the DOEN bit is not set and no interrupt is output.

To perform filter processing without accumulation/decimation and output an interrupt, set the PRCSA bit in CTLBCHn to 10_B (decimation) and set the decimation count to 0.

The DOEN flag in STCHn is automatically cleared to 0 by reading the output register. If another calculation result is written to the output data register with the DOEN flag in STCHn set to 1, the value in the output data register is overwritten. At this time, the DOOW bit in STCHn is set to 1.

The DOEN bit in STCHn is cleared to 0 by reading DOCHn (output data register) from the peripheral bus or setting the CLRDOOW bit in CLRSTCHn to 1.

28.4.13 Status Register

The DFE has a status register for each channel. The status register indicates the status shown below. Each flag except VALID can be cleared by using the corresponding clear status register. Only the DOEN flag can be automatically cleared by reading the output data register in addition to the clear status register.

Table 28.30 Status Register

Name	Description
VALID	Input Data Register Valid This bit is set to 1 when the input data register contains a valid value. This bit is cleared to 0 by a reset or at completion of accumulation/decimation processing.
CND	Condition Match This bit is set to 1 at the end of PH processing when PH is specified. This bit is also set to 1 when the comparison calculation result is true when comparison calculation is specified.
DOEN	Output Data Register Enable This bit is set to 1 when a calculation result with CTLBCHn.PRCSA = 01_B (accumulation) or CTLBCHn.PRCSA = 10_B (decimation) is written to an output register.
DIOW	Input Data Overwrite Error This bit is set to 1 when the input data register is overwritten during processing for the relevant channel. This bit is also set to 1 when the input data register in processing is overwritten by another input data with the same tag.
DOOW	Output Data Overwrite Error This bit is set to 1 when another calculation result is written while the output data register contains valid data.
MER	Multiplication Error This bit is set to 1 when calculation 8000_H*8000_H is performed in the filtering circuit.
GER	Guard Error This bit is set to 1 when a rounding error (48 bits to 32 bits) occurs in the filtering circuit.
CER	Cascade Rounding Error This bit is set to 1 when a rounding error (32 bits to 16 bits) occurs during execution of cascade input.
AER	Absolute Value Error This bit is set to 1 when absolute value calculation of 8000 0000_H is performed during absolute value calculation in the accumulation circuit.

28.4.14 Error Mask Registers

Functions of error mask registers are shown below. Masked error factors are excluded from error interrupt request conditions.

When the multiplication error mask flag is set to 1 in channel 0, for example, the multiplication error flag is not set to 1 even if a multiplication error occurs in channel 0. No error interrupt request is output in this case.

Table 28.31 Error Mask Registers

Flag	Description
MSKDIOW	Input data error (DIOW) mask bit
MSKDOOW	Output data error (DOOW) mask bit
MSKMER	Multiplication error (MER) mask bit
MSKGER	Guard error (GER) mask bit
MSKAER	Absolute value error (AER) mask bit
MSKCER	Cascade rounding error (CER) mask bit

28.4.15 Interrupt Requests

The DFE has 16 interrupt requests for output data for each channel, 10 condition match interrupt requests for each channel, and an error interrupt request.

28.4.15.1 Output Data Interrupt Requests

When a calculation result with CTLBCHn.PRCSA = 01_B (accumulation) or CTLBCHn.PRCSA = 10_B (decimation) is written to an output register, an output data interrupt request is output. (When the PRCSA is 00_B, a calculation result is written to an output data register but no interrupt is output.)

Interrupt requests corresponding to each channel can be output.

Interrupt request output can be controlled by using the IEO bit in the control register A.

Figure 28.41 shows the relationship between the DOEN bit in STCHn and an output data interrupt request.

Writing a DFE processing result with CTLBCHn.PRCSA = 01_B (accumulation) or CTLBCHn.PRCSA = 10_B (decimation) to an output data register sets the DOEN bit in STCHn to 1. The DOEN bit in STCHn is cleared to 0 by reading the DOCHn register by the software or by setting the CLRDOEN bit in CLRSTCHn to 1.

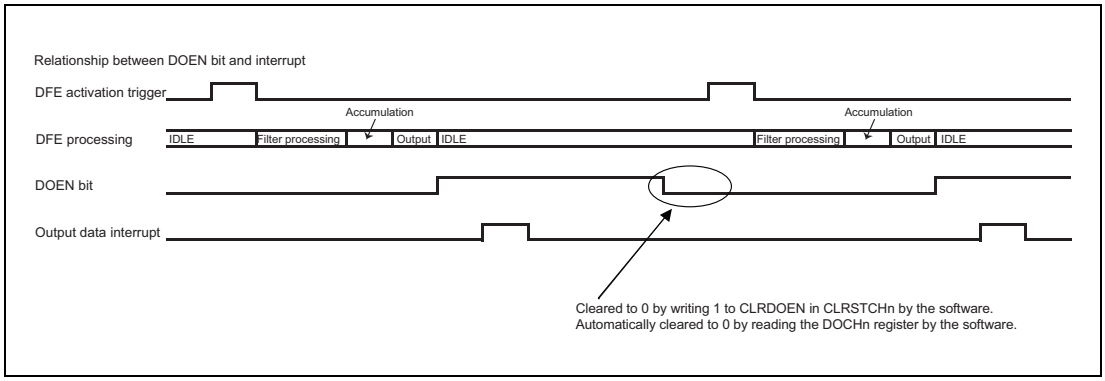


Figure 28.41 Relationship between DOEN Bit and an Interrupt

When the DOEN bit in STCHn is not cleared by the software, an output data interrupt request is not output even if data is rewritten to the output data register. If the output data register value is overwritten, the DOOW bit in STCHn is set to 1.

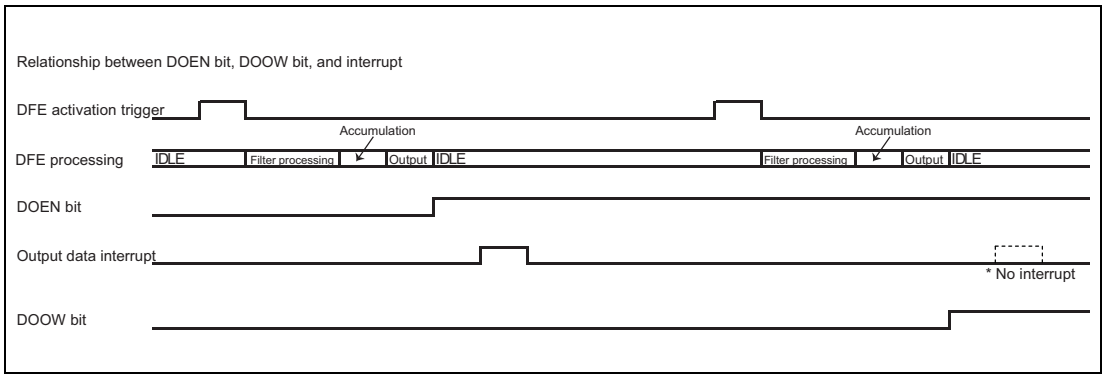


Figure 28.42 Relationship among DOEN Bit, DOOW Bit, and an Interrupt

28.4.15.2 Condition Match Interrupt Request

When the PH end trigger flag is enabled during PH processing, a condition match interrupt request indicating completion of PH processing is output. When comparison calculation result is true during comparison calculation, a condition match interrupt request is output. These requests share an interrupt. This interrupt request is given to channels 0 to 9.

Interrupt request output can be controlled by using the IEC flag in the control register A.

Figure 28.43 shows the relationship between the CND bit in the status register (STCHn (n = 0 to 9)) and condition match interrupt request. When the PH end flag is enabled, a condition match interrupt request is issued in the same timing.

The DISB bit (PH processing disable bit) in CTLBCHn disables PH processing in the same way as the enabling of the PH end flag, but does not output a condition match interrupt. In addition, even if a PH end trigger is input with PH processing, no condition match interrupt is output.

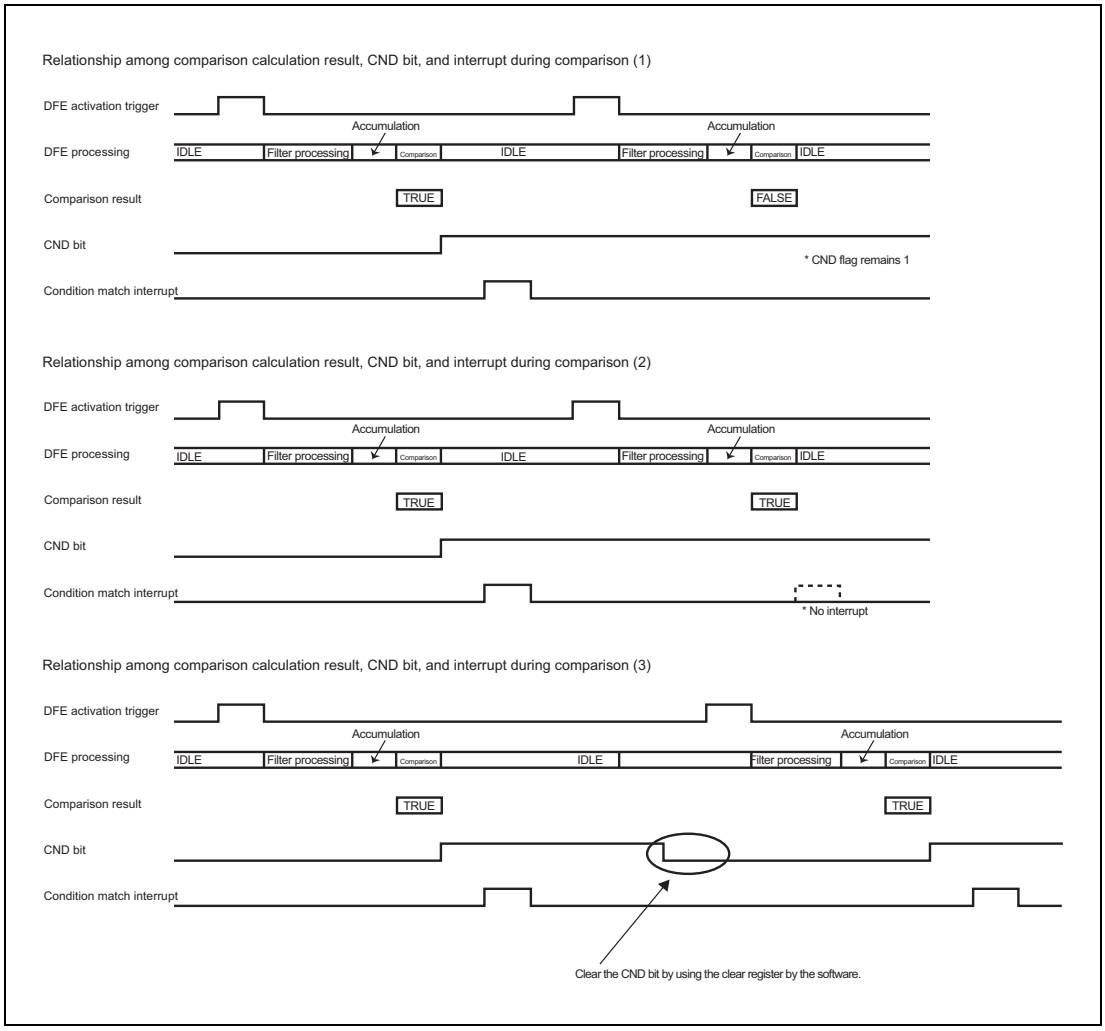


Figure 28.43 Relationship between CND Bit and Condition Match Interrupt Request

NOTE

n = 0 to 9

28.4.15.3 Error Interrupt Requests

An interrupt is generated as logical OR of errors of all channels. Error interrupt request output can be controlled for each channel.

For example, in case error interrupt requests for channel 0 to channel 7 are enabled and error interrupt requests for channel 8 to channel 15 are masked, if an error occurs in any of channels 0 to 7, an error interrupt request is output. However, if an error occurs in any of channels 8 to 15 in this state, no error interrupt request is output.

The following figure shows the relationship between error bit and error interrupt request. Since an error interrupt is generated as logical OR of error factors, another error interrupt due to a new error factor is not output while an error interrupt request is already present. After an error interrupt was output and the error bit has been cleared by the clear register, if the error bit is set again, an error interrupt is output.

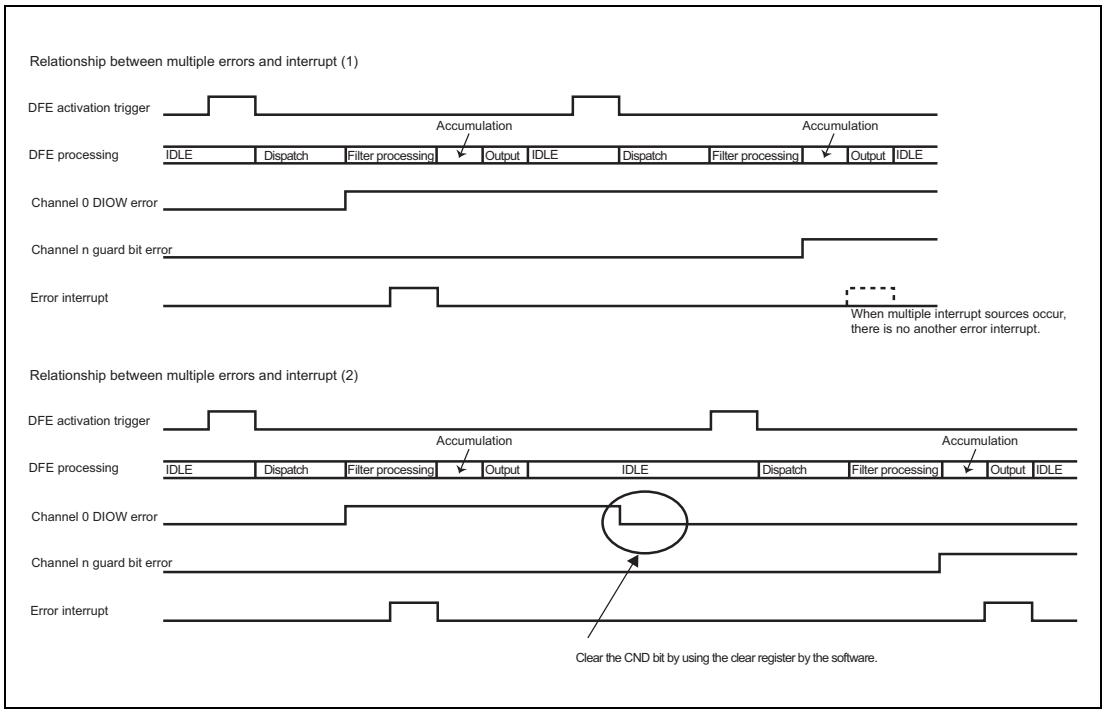


Figure 28.44 Relationship between Error Bit and Error Interrupt

28.5 Notes and Restrictions

28.5.1 Processing Time

28.5.1.1 Processing Time for One Channel

The DFE performs a single-time processing of FIR or IIR in a maximum of 0.8 µs when there is no memory access competition. If memory access competition occurs, the processing time is a maximum of 1.1 µs.

28.5.1.2 Termination Processing Time

When the EN bit in CTLACH is set to 0, the DEF performs processing for all the input data that has been accepted and terminates processing. The maximum processing time up to the termination is obtained by “the number of channels for which input data has been accepted (number of channels for which processing has not been done) x processing time”. For example, if five channels have not been processed, the termination processing time is a maximum of 5.5 µs.

28.5.2 DFE Activation Input Interval

An interval of DFE activation signal inputs should be a minimum of 0.8 µs if there is no memory access competition or a minimum of 1.1 µs if memory competition occurs.

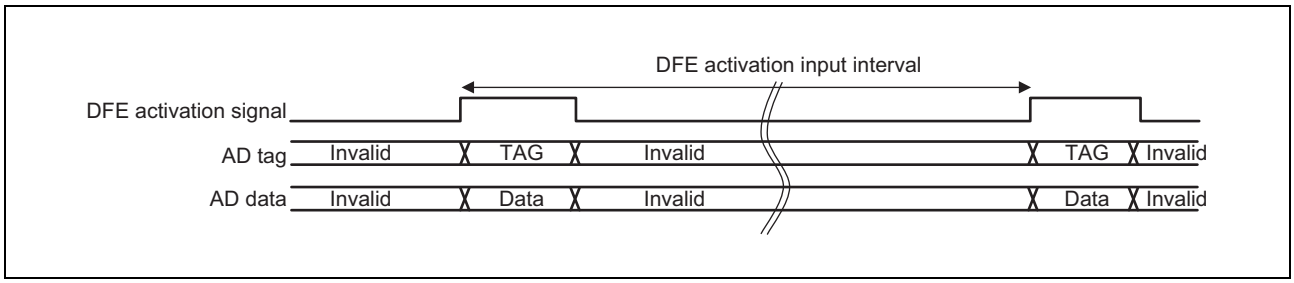


Figure 28.45 DFE Activation Input Interval (AD Input)

28.5.3 DFE Processing Time

The following two tables list the DFE processing time, i.e. the cycle from inputting a startup trigger from the AD to getting the next startup trigger on the same channel acceptable.

Table 28.32 Processing Time of FIR and IIR (Normal Cases)

		Time
FIR (normal case)	8TAP	240 ns
	16TAP	300 ns
	24TAP	350 ns
	32TAP	400 ns
IIR (normal case)	Secondary biquad 1 stage	310 ns
	Secondary biquad 2 stages	400 ns
	Secondary biquad 3 stages	490 ns

Table 28.33 Processing Time of FIR and IIR (with Initialization)

		Time
FIR (with initialization)	8TAP	310 ns
	16TAP	410 ns
	24TAP	520 ns
	32TAP	630 ns
IIR (with initialization)	Secondary biquad 1 stage	400 ns
	Secondary biquad 2 stages	570 ns
	Secondary biquad 3 stages	750 ns

Access by the CPU to the coefficient memory or data memory during DFE processing increases the total time for processing. When the frequency of the DFE operating clock is 80 MHz, the processing time increases by 25 ns per memory access.

28.5.4 Trigger Input

The following figure shows an example of supposed input of timer triggers 0 to 3 and a software trigger. When timer trigger 0 is used, for example, it is supposed that the DFE activation signal is input once or more times and timer trigger 0 is input again after the trigger flag function is enabled by the timer trigger 0 input.

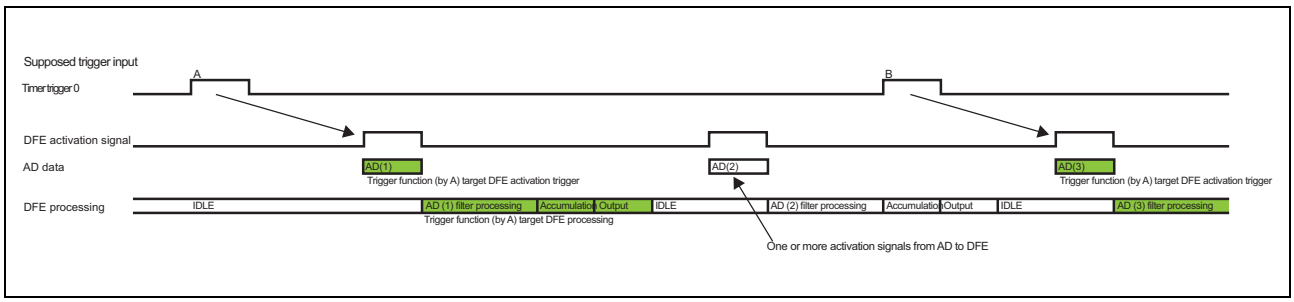


Figure 28.46 Example of Trigger Input

When a timer trigger or a software trigger is input, the corresponding flag is enabled on completion of the DFE processing. Then the flag is cleared to accept the next trigger. However, the latter trigger is not enabled when it is input before the former trigger flag is completely cleared. Therefore, successive inputs of a timer trigger or a software trigger are prohibited.

In the example below, the trigger function by B is not enabled in some cases.

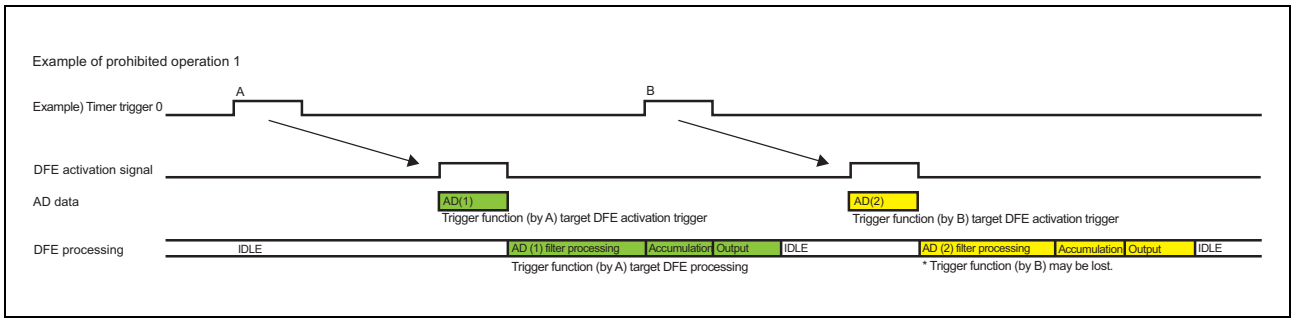


Figure 28.47 Operation in which the Same Trigger is Prohibited 1

Before a timer trigger or a software trigger is input and the DFE activation signal is input, the same timer trigger or software trigger must not be input again. In the example below, the trigger function by A and the trigger function by B are enabled only for AD (2).

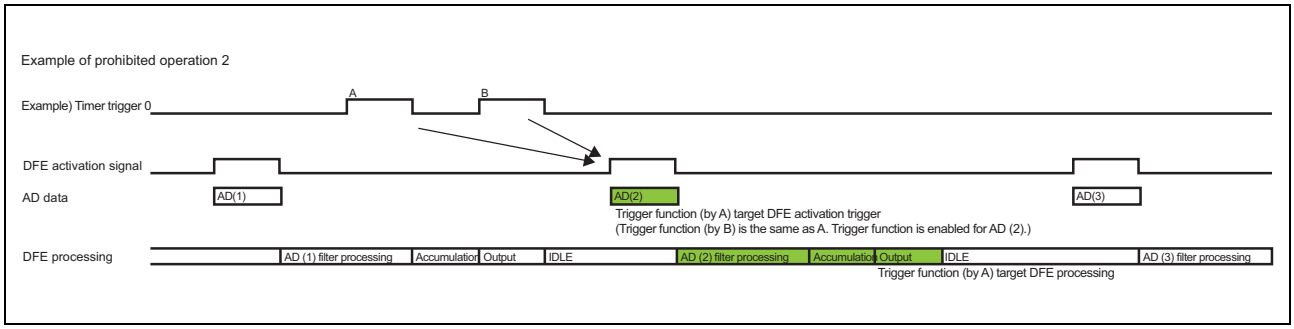


Figure 28.48 Operation in which the Same Trigger is Prohibited 2

28.5.5 Channel Tag and AD Tag

If a tag value (AD tag) that the AD inputs to the DFE and a tag value (channel tag) that the software sets for the DFE are different, correct processing is disabled. For example, the AD inputs an AD tag value of 0001_B to the DFE and there is no channel with a DFE's channel tag value 0001_B, the DFE does not perform filter processing.

28.5.6 Input Data Format and Calculation Restrictions

Restrictions on available calculations are provided for the DFE. For details, see **Section 28.4.6, Setting Control Registers**.

28.5.7 Restrictions on Cascade Processing

Cascade processing is a function to round 32-bit accumulation circuit output data to 16-bit data to be input data of other channels. The following restrictions are provided for cascade processing. Observe these restrictions together with the descriptions in **Section 28.4.6, Setting Control Registers**.

1. When cascade is enabled, do not set the same value for cascade tag and channel tag in the same channel. Assign a channel that is different from the channel for which cascade is specified as a cascade destination channel. If the same tag value is set, the filter processing result is input to the same channel again and calculation is repeated.
2. When cascade is enabled, the data format of cascade enabled channels must be the same as the data format of cascade destination channels. For example, when the fixed-point input format is specified for cascade enabled channels, also specify the fixed-point input format for cascade destination channels.
3. Cascade input can be enabled for up to 15 channels in 16 channels. If cascade input is enabled for all channels, calculation is repeated for filter calculation results that are input in cascade.
4. The CAEN bits in the control register A (CTLACHn) must not be set to 01_B.
5. When PH processing and comparison processing are enable, PH processing and comparison processing are performed for 32-bit data.
6. No floating-point conversion result can be a cascade input.
7. When enabling the cascade function of two or more channels and selecting a single cascade destination channel, pay attention to the priority of channels. Specify cascade channels so that higher priority is given to cascade destination channels than cascade enabled channels. To enable the cascade function of channel 1 and channel 2 and select the same cascade destination for channel 1 and channel 2, for example, specify channel 0 as a cascade destination channel.

28.5.8 Operation when a Channel is Disabled

While the channel enable bit (CTLACHn.EN) is set to 0, filter processing is not activated even if the AD tag value that is input from the AD is equal to the channel tag value.

In addition, timer trigger input of the trigger flag function specified for the channel and software trigger input are not accepted.

When the channel enable bit (CTLACHn.EN) is set to 0 from 1, data to be processed that was input before this bit is set to 0 is processed.

At this time, if a trigger for the trigger flag function is input before the channel enable bit is set to 0, the trigger flag function is enabled for the data to be processed.

When the channel enable bit (CTLACHn.EN) is 1, input trigger for the trigger flag function is held until AD data is input and the trigger flag function is enabled for the data to be processed (trigger held).

After the channel enable bit (CTLACHn.EN) is set to 0 from 1, the held trigger is cleared if there is no data to be processed.

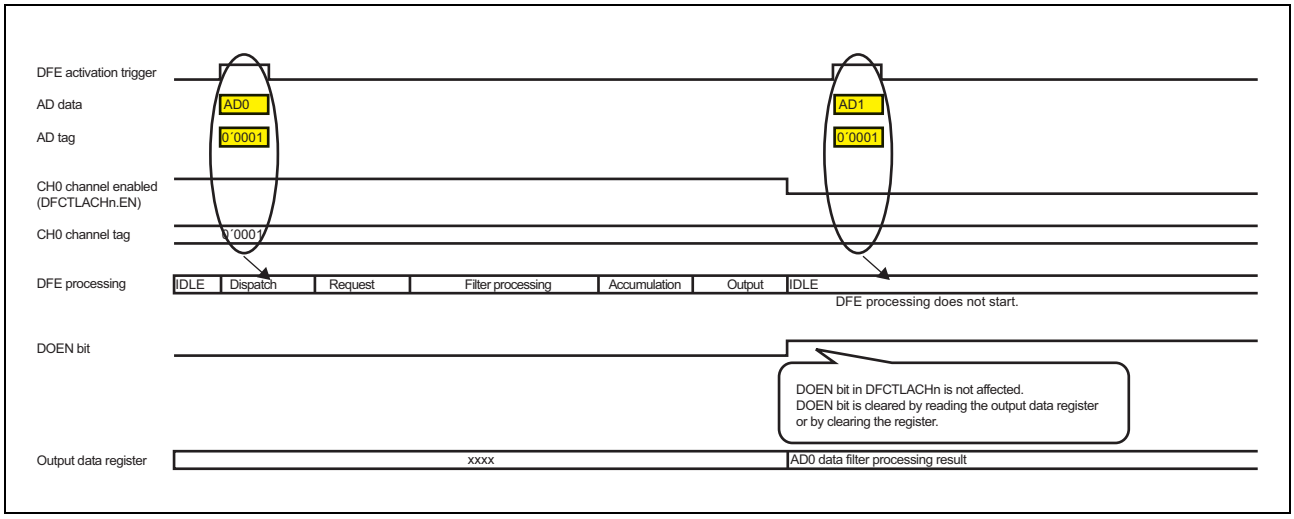


Figure 28.49 EN Bit and DOEN Bit

Even if the channel enable bit (CTLACHn.EN) is cleared to 0, the status register is not cleared. Clear the status register by the software as needed.

If the channel enable bit is set to 1 from 0 to restart the DFE processing.

- Accumulation/decimation counter starts at 0 in the same way as the first DFE processing after a reset.
- Comparison with the initial value register value is performed during the first PH processing, and comparison with the PH result register value is performed during the subsequent processing.

28.5.9 Restrictions on Memory Access

8-bit or 16-bit write access to the coefficient memory is prohibited. Write data to the coefficient memory in 32 bits. 8-bit write access to data memory 0 and data memory 1 is prohibited. Write data to data memory 0 and data memory 1 in 16 bits.

If a prohibited access is made, data may be broken. In the case of the coefficient memory, word data (32 bits) including a byte-write address may be broken. In the case of data memory 0 and data memory 1, half-word data (16 bits) including a byte-write address may be broken.

28.5.10 Restrictions on Trigger Setting Registers

A software trigger must not be used to set both the accumulation/decimation initialization and the accumulation/decimation disable setting. Operation is not guaranteed if this setting is made.

A software trigger must not be used to set both the PH initialization and the PH end setting. Operation is not guaranteed if this setting is made.

28.5.11 Setting of PH Processing Disable Bit and Accumulation/Decimation Processing Disable Bit

When the EN bit in CTLACHn and VALID bit in STCHn are both 0, the DISB bit (PH processing disable bit) in CTLBCHn and DISA bit (accumulation/decimation processing disable bit) in CTLBCHn can be set by software.

To disable PH processing by writing 1 to the DISB bit in CTLBCHn, set the PRCSB bit in CTLBCHn to 01_B (for PH processing). To disable accumulation/decimation processing by writing 1 to the DISA bit in CTLBCHn, set the PRCSA bit to 01_B (for accumulation processing) or 10_B (for decimation processing).

Section 29 Safety

29.1 Overview

This section describes the failure detection functions provided to detect the LSI failures in the early stage. Here, the failures include both the recoverable transient failures such as software errors of a memory and the unrecoverable permanent failures.

This microcontroller was developed to meet the requirements for a Safety Element out of Context (SEooC) as described in the ISO26262. Contact our sale office for the details regarding the development process and safety organization.

The following lists the failure detection functions provided by this LSI.

ECC and EDC

Detect failures of memories and data transfer paths; correct some types of failures.

Lockstep

Detects failures of the CPU1 in the early stage.

NOTE

The lockstep function is disabled during debugging.

Failures are not detected even when the failure detection function is used.

Memory Protection

Detects erroneous access to memories and peripheral circuits to protect the data in these elements against erroneously access.

MISG

Monitors write access to a specific address by the CPU or PCU, generates the signatures based on the written data, and automatically compares the generated signatures with each other.

Clock Monitor

Monitors the clock operation to detect abnormal operation.

BIST

Detects failures of the failure detection function itself.

Error Control Module (ECM)

Monitors various failure detection states in the LSI and defines the operation to be carried out upon failure detection.

29.2 ECC and EDC

29.2.1 Overview

29.2.1.1 ECC

This product incorporates ECC for the following memories. The ECC enables detection and correction of errors of the data retained in the memories. The ECC also enables detection and correction of errors produced between the ECC encoder and memories; and memories and ECC decoder.

Table 29.1 ECC Overview

Applicable Memory	Applicable Data Width [bits]	Operation upon Error Detection				Failure Insertion
		Detection/ Correction	Notice to ECM	Error Status	Address Capture	
Code flash	128	SEC-DED	Possible	Possible	Possible	Possible
Data flash Local RAM (CPU1) Local RAM (PCU) Global RAM	32	SEC-DED	Possible	Possible	Possible	Possible
Instruction cache (data)	64	SEC-DED	Possible	Possible	Possible	Possible
Instruction cache (tag)	32	SED-DED	Possible	Possible	Possible	Possible
RAM for DTS	32	SEC-DED	Possible	Possible	Possible	Possible
Peripheral RAM (32 bits)	32	SEC-DED	Possible	Possible	Possible	Possible
Peripheral RAM (16 bits)	16	SEC-DED	Possible	Possible	Possible	Possible

Applicable Data Width

ECC encoding is applied to the data of the shown width. If narrower data is to be written, the following processes are required. Here, ECC is also checked when data is read out in (1).

(1) Read data to which ECC encoding is applied, including data to be rewritten.

(2) Replace data to be rewritten.

(3) Write back the data generated in (2).

Detection/Correction

SEC-DED: 1-bit errors can be detected and corrected, and 2-bit errors can only be detected.

SED-DED: 1-bit errors and 2-bit errors can only be detected.

Notice to ECM

An error detected can be notified to the ECM (error control module).

Error Status

The status of an error detected is retained.

Address Capture

The address of an error detected is retained.

Failure Insertion

An ECC error can be intentionally caused to enable self-diagnosis of the ECC decoder operation.

29.2.1.2 Address Parity

This product incorporates address parity for the following memories. The address parity enables detection of errors during address decoding. The EDC also enables detection of errors produced at addresses between the parity encoder and memories.

Table 29.2 Address Parity Overview

Applicable Memory	Parity Bit	Notice to ECM	Error Status	Address Capture	Failure Insertion
Code flash	1 bit	Possible	Possible	Possible	Possible
Local RAM (CPU1)	2 bits*1	Possible	Possible	Possible	Possible
Global RAM	1 bit	Possible	Possible	Possible	Possible

Note 1. The parity bits corresponding to the written address are written to two locations in the memory. When they are compared to the parity bits corresponding to the read address and errors are detected at both of the two parity bits stored in the memory, it is handled as an address parity error. When an error is detected only at one of the parity bits, it is handled as a parity bit error.

29.2.1.3 Data Parity

This product incorporates data parity for the particular data transfer. The data EDC enables detection of errors of the transferred data. For details, refer to **Section 29.2.11, Data Parity for Data Transfer Paths**.

29.2.2 Code Flash ECC and Address Parity

29.2.2.1 Overview

The code flash ECC is summarized in the table below.

Table 29.3 Overview of Code Flash ECC

Item	Description
ECC error detection and correction	ECC error detection and correction can be either enabled or disabled. When enabled, either of the following settings can be selected. • ECC error detection and correction are carried out (2-bit error detection and 1-bit error detection and correction are carried out). • ECC error detection is carried out (2-bit error detection and 1-bit error detection are carried out). In the state after reset, detection of a 2-bit error and detection and correction of a 1-bit error will be notified while the function is enabled.
Address parity	Address parity check can be either enabled or disabled. Address parity is checked during data read. In the state after reset, this function is enabled.
Error notification	Upon occurrence of an ECC error or parity error, the ECM is notified of the error. **ECC Error:** • Error notification can be either enabled or disabled upon detection of a 2-bit ECC error. • Error notification can be either enabled or disabled upon detection of a 1-bit ECC error. In the state after reset, error notification is enabled upon detection of a 2-bit ECC error, and error notification is disabled upon detection of a 1-bit ECC error. **Parity Error:** • Error notification can be either enabled or disabled upon detection of an address parity error. In the state after reset, error notification is enabled upon detection of an address parity error. The error notification signal is issued to the ECM, where a 2-bit ECC error and an address parity error are handled as one source, and a 1-bit ECC error is handled as one source.
Error status	A status register is provided, which indicates the statuses of 2-bit ECC error detection, 1-bit ECC error detection, and address parity error detection. If an error occurs while no error status flag is set, the corresponding status is set. The error status flag can be cleared using the clear register.
Address capture	If an ECC error or a parity error occurs while no error status flag is set, the address at which the associated error has occurred is captured. The address is captured when a 2-bit ECC error, a 1-bit ECC error, or an address parity error is detected. The error status flag serves as the enable bit of the capture address.
Self-diagnosis	Data in the ROM and the ECC and address parity bits can all be read directly. Desired values can also be written to the ROM and to the ECC and address parity bits.
Others	The ECM can initiate a transition to the safe state in response to the detection of a 2-bit ECC error during the fetching of an instruction.

The ECC decoder and address parity generator are provided each for the read ports (CPU1, PCU, and interconnect) connected to the code flash interface. The address parity checker is provided in the access controller near the code flash. For details, see **Figure 29.1**.

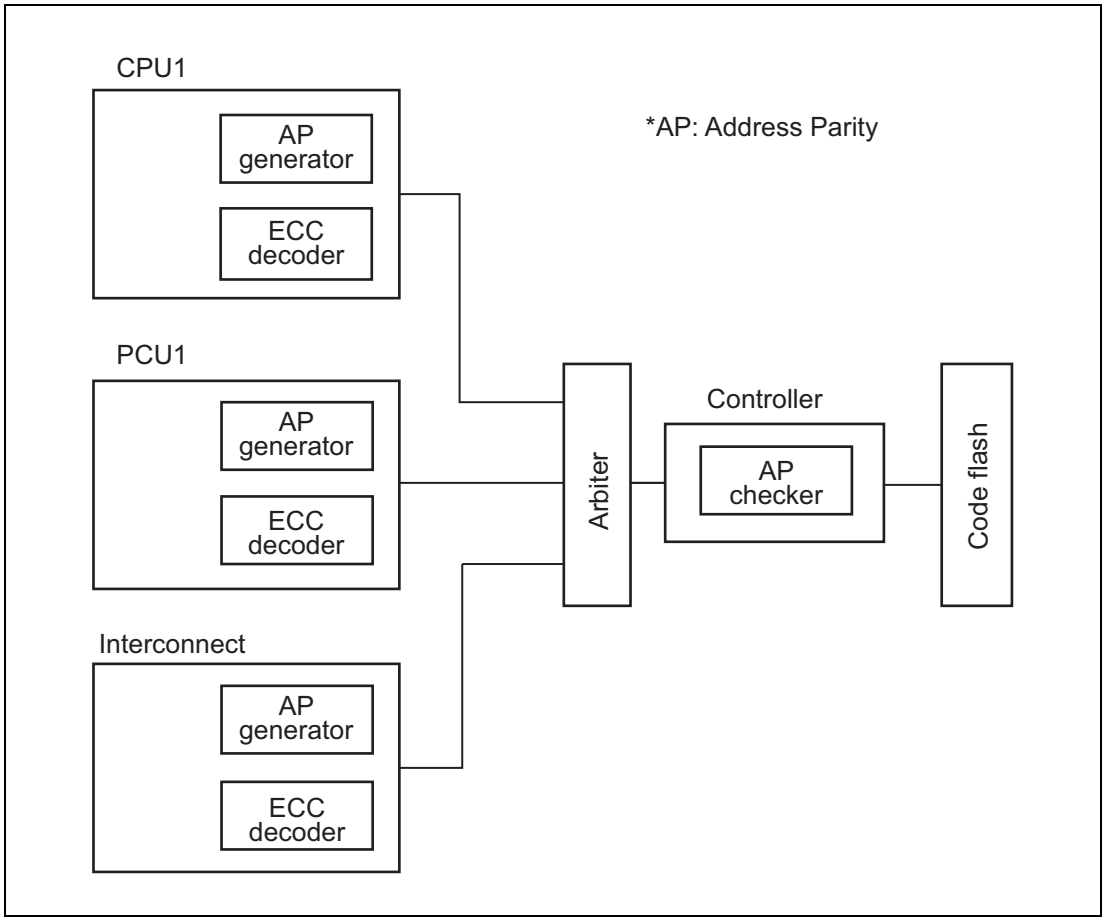


Figure 29.1 ECC and Address Parity of Code Flash

29.2.2.2 List of Registers

Table 29.4 List of Registers

Address	Symbol*1	Register Name	R/W	Value after Reset	Access Size
FFC6 2000_H	CFAPCTL	Code flash address parity control register	R/W	0000 0000_H	16/32
FFC6 2200_H	CFECCCTL_VCI	Code flash ECC control register (VCI)	R/W	0000 0000_H	16/32
FFC6 2204_H	CFERRINT_VCI	Code flash error information control register (VCI)	R/W	0000 0006_H	8/16/32
FFC6 2208_H	CFSTCLR_VCI	Code flash status clear register (VCI)	W	0000 0000_H	8/16/32
FFC6 $220C_H$	CFOVFSTR_VCI	Code flash error count overflow status register (VCI)	R	0000 0000_H	8/16/32
FFC6 2210_H	CF1STERSTR_VCI	Code flash 1st error status register (VCI)	R	0000 0000_H	8/16/32
FFC6 2250_H	CF1STEADR0_VCI	Code flash 1st error address register (VCI)	R	0000 0000_H	8/16/32
FFC6 2350_H	CFSTSTCTL_VCI	Code flash sub test control register (VCI)	R/W	0000 0000_H	16/32
FFC6 2400_H	CFECCCTL_PE1	Code flash ECC control register (PE1)	R/W	0000 0000_H	16/32
FFC6 2404_H	CFERRINT_PE1	Code flash error information control register (PE1)	R/W	0000 0006_H	8/16/32
FFC6 2408_H	CFSTCLR_PE1	Code flash status clear register (PE1)	W	0000 0000_H	8/16/32
FFC6 $240C_H$	CFOVFSTR_PE1	Code flash error count overflow status register (PE1)	R	0000 0000_H	8/16/32
FFC6 2410_H	CF1STERSTR_PE1	Code flash 1st error status register (PE1)	R	0000 0000_H	8/16/32
FFC6 2450_H	CF1STEADR0_PE1	Code flash 1st error address register (PE1)	R	0000 0000_H	8/16/32
FFC6 2550_H	CFSTSTCTL_PE1	Code flash sub test control register (PE1)	R/W	0000 0000_H	16/32
FFC6 2800_H	CFECCCTL_PCU	Code flash ECC control register (PCU)	R/W	0000 0000_H	16/32
FFC6 2804_H	CFERRINT_PCU	Code flash error information control register (PCU)	R/W	0000 0006_H	8/16/32
FFC6 2808_H	CFSTCLR_PCU	Code flash status clear register (PCU)	W	0000 0000_H	8/16/32
FFC6 $280C_H$	CFOVFSTR_PCU	Code flash error count overflow status register (PCU)	R	0000 0000_H	8/16/32
FFC6 2810_H	CF1STERSTR_PCU	Code flash 1st error status register (PCU)	R	0000 0000_H	8/16/32
FFC6 2850_H	CF1STEADR0_PCU	Code flash 1st error address register (PCU)	R	0000 0000_H	8/16/32
FFC6 2950_H	CFSTSTCTL_PCU	Code flash sub test control register (PCU)	R/W	0000 0000_H	16/32

Note 1. The registers with symbols "_VCI", "_PE1", and "_PCU" as suffixes are provided to the particular ECC controllers: the registers with "_VCI" are provided to the ECC controller for access from the system interconnect 1 to the code flash, the registers with "_PE1" are provided to the ECC controller for access from the CPU1, and the registers with "_PCU" are provided to the ECC controller for access from the PCU.

29.2.2.3 Details of Registers

(1) CFAPCTL — Code Flash Address Parity Control Register

CFAPCTL enables or disables address parity check. Set the PROT[1:0] bits to 01_B when writing to this register.

This register is initialized by an internal reset or an external reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PROT[1:0]		—	—	—	—	—	—	—	—	—	—	—	APTESTB	APTESTA	APARIDIS
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Table 29.5 CFAPCTL Register Contents

Bit Position	Bit Name	Function
31 to 16	—	Reserved These bits are always read as 0. The write value should also be 0.
15, 14	PROT[1:0]	These bits enable writng to this register. The read value is 0.
13 to 3	—	Reserved These bits are always read as 0. The write value should also be 0.
2	APTESTB	Address Parity Checker (Bank B) Test Sets the address parity checker to test mode. When APTESTB = 1, the parity generated by the address parity generator is inverted.
1	APTESTA	Address Parity Checker (Bank A) Test Sets the address parity checker to test mode. When APTESTA = 1, the parity generated by the address parity generator is inverted.
0	APARIDIS	Address Parity Check Disable Enables or disables address parity check by the address parity circuit. 0: Enables address parity check. 1: Disables address parity check.

(2) CFECCCTL_VCI/_PE1/_PCU — Code Flash ECC Control Register

CFECCCTL enables or disables ECC error detection and correction and 1-bit error correction. Set the PROT[1:0] bits to 01_B when writing to this register.

This register is initialized by an internal reset or an external reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PROT[1:0]		—	—	—	—	—	—	—	—	—	—	—	—	SECDIS	ECCDIS
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Table 29.6 CFECCCTL Register Contents

Bit Position	Bit Name	Function
31 to 16	—	Reserved These bits are always read as 0. The write value should also be 0.
15, 14	PROT[1:0]	These bits enable to write to this register. The read value is 0.
13 to 2	—	Reserved These bits are always read as 0. The write value should also be 0.
1	SECDIS	1-Bit Error Correction Disable Enables or disables 1-bit error correction when ECC error detection and correction are enabled. 0: Enables correction of the 1-bit error detected. 1: Disables correction of the 1-bit error detected.
0	ECCDIS	ECC Disable Enables or disables ECC error detection and correction. 0: Enables ECC error detection and correction. 1: Disables ECC error detection and correction.

(3) CFERRINT_VCI/_PE1/_PCU — Code Flash Error Information Control Register

CFERRINT enables or disables generation of the error notification signal to the ECM upon detection of a 2-bit ECC error, a 1-bit ECC error, or an address parity error.

CFERRINT is initialized by an internal reset or an external reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	APEIE	DEDIE	SEDIE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Table 29.7 CFERRINT Register Contents

Bit Position	Bit Name	Function
31 to 3	—	Reserved These bits are always read as 0. The write value should also be 0.
2	APEIE	Address Parity Error Notification Enable Enables or disables generation of the error notification signal upon detection of an address parity error when address parity check is enabled. 0: Disables notification of the address parity error. 1: Enables notification of the address parity error.
1	DEDIE	2-Bit ECC Error Notification Enable Enables or disables generation of the error notification signal upon detection of a 2-bit error when ECC error detection and correction are enabled. 0: Disables notification of the 2-bit ECC error. 1: Enables notification of the 2-bit ECC error.
0	SEDIE	1-Bit ECC Error Notification Enable Enables or disables generation of the error notification signal upon detection of a 1-bit error when ECC error detection and correction are enabled. 0: Disables notification of the 1-bit ECC error. 1: Enables notification of the 1-bit ECC error.

(4) CFSTCLR_VCI/_PE1/_PCU — Code Flash Status Clear Register

CFSTCLR clears the error flags in the error status register (CF1STERSTR), the overflow flag in the error count overflow status register (CFOVFSTR), and the error address register (CF1STEADR). CFSTCLR is a write-only register and is always read as 0.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	STCLR0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	W

Table 29.8 CFSTCLR Register Contents

Bit Position	Bit Name	Function
31 to 1	—	Reserved These bits are always read as 0. The write value should also be 0.
0	STCLR0	Error Overflow Flag Clear Writing 1 to this bit clears the APEF0, DEDF0, and SEDF0 flags in CF1STERSTR; ERROVF0 flag in CFOVFSTR; and CF1STEADR0.

(5) CFOVFSTR_VCI/_PEI/_PCU — Code Flash Error Count Overflow Status Register

CFOVFSTR monitors occurrence of error count overflow. If the second error occurs after the first error (= while any of the error flags in the error status register is set), the flag in this register is set. However, if the second error is identical to the first error (both the source and address are same), this flag is not set. ERROVF is cleared by an internal reset, an external reset, or setting the STCLR bit to 1 in CFSTCLR.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ERROVF0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 29.9 CFOVFSTR Register Contents

Bit Position	Bit Name	Function
31 to 1	—	Reserved These bits are always read as 0. The write value should also be 0.
0	ERROVF0	Error Overflow Flag This flag is set if the second error occurs while any of the error flags (APEF0, DEDF0, and SEDF0) in the error status register is set, except when both of the error address and source of the second error are the same as those of the first error.

(6) CF1STERSTR_VCI/_PE1/_PCU — Code Flash 1st Error Status Register

CF1STERSTR monitors occurrence of the first error. An error status flag is set if an error occurs while the error flag is 0. Note, however, that the corresponding error flag is also set if a 2-bit ECC error or an address parity error occurs in subsequent access as long as only the 1-bit ECC error flag is set at the time.

If more than one error is detected for a given memory access, all the corresponding error flags are set. CF1STERSTR is cleared by an internal reset, an external reset, or setting the STCLR bit to 1 in CFSTCLR.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	APEF0	DEDF0	SEDF0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 29.10 CF1STERSTR Register Contents

Bit Position	Bit Name	Function
31 to 3	—	Reserved These bits are always read as 0. The write value should also be 0.
2	APEF0	Address Parity Error Monitor Flag Clearing condition: Cleared to 0 by a reset or by setting the STCLR0 bit to 1 in CFSTCLR. Setting condition: An address parity error has occurred while the error flags DEDF0 and APEF0 are 0.
1	DEDF0	2-Bit ECC Error Monitor Flag Clearing condition: Cleared to 0 by a reset or by setting the STCLR0 bit to 1 in CFSTCLR. Setting condition: A 2-bit ECC error has occurred while the error flags DEDF0 and APEF0 are 0.
0	SEDF0	1-Bit ECC Error Monitor Flag Clearing condition: Cleared to 0 by a reset or by setting the STCLR0 bit to 1 in CFSTCLR. Setting condition: A 1-bit ECC error has occurred while the error flags DEDF0, SEDF0, and APEF0 are 0.

(7) CF1STEADR0_VCI/_PE1/_PCU — Code Flash 1st Error Address Register

CF1STEADR0 holds the address at which an error has occurred.

The error address is set if an error occurs while all the error flags are 0 in CF1STERSTR. The address is updated if a 2-bit ECC error or an address parity error occurs while the 1-bit ECC error flag is set as the first error. Once a 2-bit ECC error or an address parity error occurs, the address is not updated.

EADR[24:4] of this register correspond to bits [24:4] of the real address. The real address can be calculated by appending the higher-order address bits [31:25] as a base address.

CF1STEADR0 is cleared by an internal reset, an external reset, or setting the STCLR bit to 1 in CFSTCLR.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	EADR[24:16]								
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	EADR[15:4]												—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 29.11 CF1STEADR0 Register Contents

Bit Position	Bit Name	Function
31 to 25	—	Reserved These bits are always read as 0. The write value should also be 0.
24 to 4	EADR[24:4]	1st Error Address These bits monitor the address of the first error. The error address is set if an error occurs while all the error flags are 0 in CF1STERSTR. The address is updated if a 2-bit ECC error or an address parity error occurs while the 1-bit ECC error flag is set as the first error. Once a 2-bit ECC error or an address parity error occurs, the address is not updated.
3 to 0	—	Reserved These bits are always read as 0. The write value should also be 0.

(8) CFSTSTCTL_VCI/PE1/PCU — Code Flash Sub-Test Control Register

CFSTSTCTL is used for the ECC test (self-diagnosis). This register is dedicated for the code flash. After ECC test mode is enabled by setting ECCTST = 1, the ECC and address parity bits can be read directly. Set the PROT[1:0] bits to 01_B when writing to this register.

This register is initialized by an internal reset or an external reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PROT[1:0]		—	—	—	—	—	—	—	—	—	—	—	—	—	ECCTST
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 29.12 CFSTSTCTL_VCI/PE1/PCU Registers Contents

Bit Position	Bit Name	Function
31 to 16	—	Reserved These bits are always read as 0. The write value should also be 0
15, 14	PROT[1:0]	These bits enable to write to this register. The read value is 0.
13 to 1	—	Reserved These bits are always read as 0. The write value should also be 0
0	ECCTST	ECC Test After ECC test mode is enabled by setting ECCTST = 1, the ECC and address parity bits can be read directly.

Correctly reading instructions from the code flash access port is not possible while ECC test mode is selected (ECCTST = 1). While the access port for the CPU or PCU is set to test mode (including during changes to the value of the ECCTST bit), the CPU or PCU must run a program from the local RAM or global RAM and must not fetch instructions from the code flash memory.

The CPU has a small data buffer. If an old value remains in this buffer, the correct value cannot be read even when the ECCTST bit is switched. When switching the ECCTST bit, be sure to clear the data buffer. For how to clear the data buffer, see **Section 3, CPU System**.

From the code flash access port with ECC test mode selected, access must be made by reading 4 bytes aligned to 16n address. The results of code flash reading are as follows:

Table 29.13 Results of Code Flash Reading

Bit Position	Contents
31 to10	Always 0
9	Address parity bit
8 to 0	ECC bits

29.2.2.4 Test Function

Through appropriate register setting, the code flash data, ECC bits, and address parity bit can be read out.

(1) Reading Code Flash Data

(1) Set the ECCDIS bit in the code flash ECC control register to 1 to disable ECC error detection and correction.

(2) When ECCDIS = 1, neither error detection nor correction proceeds when the code flash is read; the data output from the code flash is read unchanged.

How to exit this test mode:

(1) Set the ECCDIS bit in the code flash ECC control register to 0 to enable ECC error detection and correction.

(2) Reading the ECC and Address Parity Bits

(1) Set the ECCDIS bit in the code flash ECC control register to 1 to disable ECC error detection and correction.

(2) Set the ECCTST bit in the code flash sub-test control register to 1 to set test mode.

(3) When the code flash is read, the ECC and address parity bits are read instead of the code flash data.

How to exit this test mode:

(1) Set the ECCDIS bit in the code flash ECC control register to 0 to enable ECC error detection and correction.

(2) Set the ECCTST bit in the code flash sub-test control register to 0 to set normal mode.

(3) Self-Diagnosis

Self-diagnosis of the ECC decoder and address parity decoder for the access ports is possible by writing incorrect data to the code flash memory beforehand (fault injection) and then reading this data. A 1- or 2-bit ECC error or an address parity error fault can be injected by generating correct ECC bits and address parity bit once and inverting only the appropriate bits.

For details on programming of the code flash, refer to the *RH850/E1x Flash Memory User's Manual: Hardware Interface*.

(4) Self-Diagnosis of the ECC and Address Parity for Code Flash Memory

Data for generating a 1-bit ECC error, 2-bit ECC error, or address parity error has been written to the area from 0100 $A000_H$ to 0100 $A8CF_H$ in advance for the self-diagnosis of the ECC and address parity for code flash memory, and read access to the area can intentionally generate an error to allow self-diagnosis of the operation of the ECC and address parity.

The tables below lists the addresses written with the error-generating code.

Table 29.14 Code Flash Diagnosis Code

Address	Description
0100 $A000_H$ to 0100 $A08F_H$	1-bit ECC error generation (ECC code area walking)
0100 $A090_H$ to 0100 $A88F_H$	1-bit ECC error generation (data area walking)
0100 $A8B0_H$ to 0100 $A8BF_H$	2-bit ECC error generation
0100 $A8C0_H$ to 0100 $A8CF_H$	Address parity error generation

29.2.3 Data Flash ECC

29.2.3.1 Overview

The data flash ECC is summarized in the table below.

Table 29.15 Overview of Data Flash ECC

Item	Description
ECC error detection and correction	ECC error detection and correction can be either enabled or disabled. When enabled, either of the following settings can be selected. • ECC error detection and correction are carried out (2-bit error detection and 1-bit error detection and correction are carried out). • ECC error detection is carried out (2-bit error detection and 1-bit error detection are carried out). When disabled, neither error detection nor correction is carried out. In the state after reset, this function is enabled, and 1-bit error detection and correction and 2-bit error detection and notification are carried out.
Error notification	Upon occurrence of an ECC error, the ECM is notified of the error. **ECC Error:** • Error notification can be either enabled or disabled upon detection of a 2-bit ECC error. • Error notification can be either enabled or disabled upon detection of a 1-bit ECC error. In the state after reset, error notification is enabled upon detection of a 2-bit ECC error, and error notification is disabled upon detection of a 1-bit ECC error. The error notification signal is output, where a 2-bit ECC error is handled as one source, and a 1-bit ECC error is handled as one source.
Error status	A status register is provided, which indicates the statuses of 2-bit ECC error detection and 1-bit ECC error detection. If an error occurs while no error status flag is set, the corresponding status is set. The error status flag can be cleared using the clear register.
Address capture	If an ECC error occurs while no error status flag is set, the address at which the associated error has occurred is captured. The address is captured when a 2-bit ECC error or a 1-bit ECC error is detected. The error status flag serves as the enable bit of the capture address.
Self-diagnosis	Data in the ROM and the ECC bits can be read directly. Desired values can be written as ROM data and to the ECC bits.

29.2.3.2 List of Registers

(1) List of ECC Modules

Table 29.16 List of Modules

Name of ECC Modules and Register Base Address			
Master Side		Checker Side	
Module Name	Base Address <Base_addr>	Module Name	Base Address <Base_addr>
ECCEEP	FFC6 $2C00_H$	ECCEEPC	FFC6 $2E00_H$

(2) List of Registers

Each ECC module has the registers shown in the following table.

Table 29.17 List of Registers

Symbol	Register Name	R/W	Value after Reset	Address	Access Size		
					8	16	32
DFECCCTL	Data flash ECC control register	R/W	0000_H	<Base_addr>		√	
DFERSTR	Data flash error status register	R	00_H	<Base_addr> + 04_H	√		
DFERSTC	Data flash error status clear register	W	00_H	<Base_addr> + 08_H	√		
DFOVFSTR	Data flash error overflow status register	R	00_H	<Base_addr> + $0C_H$	√		
DFOVFSTC	Data flash error overflow status clear register	W	00_H	<Base_addr> + 10_H	√		
DFERRINT	Data flash error notification control register	R/W	02_H	<Base_addr> + 14_H	√		
DFEADR	Data flash 1st error address register	R	$0000\ 0000_H$	<Base_addr> + 18_H			√
DFTSTCTL	Data flash test control register	R/W	0000_H	<Base_addr> + $1C_H$		√	

(3) DFECCCTL — Data Flash ECC Control Register

DFECCCTL enables or disables ECC error detection and correction and 1-bit error correction.

This register is initialized by an internal reset or an external reset.

Set the PROT[1:0] bits to 01_B when writing to this register.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PROT[1:0]		—	—	—	—	—	—	—	—	—	—	—	—	SECDIS	ECCDIS
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Table 29.18 DFECCCTL Register Contents

Bit Position	Bit Name	Function
15, 14	PROT[1:0]	These bits enable to write to this register. The read value is 0.
13 to 2	—	Reserved
1	SECDIS	1-Bit Error Correction Disable Enables or disables 1-bit error correction when ECC error detection and correction are enabled. 0: Enables correction of the 1-bit error detected. 1: Disables correction of the 1-bit error detected.
0	ECCDIS	ECC Disable Enables or disables ECC error detection and correction. In the initial state, ECC error detection and correction are enabled. 0: Enables ECC error detection and correction. 1: Disables ECC error detection and correction.

(4) DFERSTR — Data Flash Error Status Register

DFERSTR monitors occurrence of errors.

DFERSTR is initialized by an internal reset, an external reset, or setting the clear bit in the data flash error status clear register.

The SEDF bit is set if a 1-bit ECC error is detected while ECC error detection and correction are enabled, and the DEDF bit is set if a 2-bit ECC error is detected.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	DEDF	SEDF
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 29.19 DFERSTR Register Contents

Bit Position	Bit Name	Function
7 to 2	—	Reserved
1	DEDF	2-Bit ECC Error Monitor Flag Clearing condition: Cleared to 0 by a reset or by setting 1 to the ERRCLR bit in DFERSTC. Setting condition: A 2-bit ECC error has occurred while the error flags DEDF and SEDF are 0.
0	SEDF	1-Bit ECC Error Monitor Flag Clearing condition: Cleared to 0 by a reset or by setting 1 to the ERRCLR bit in DFERSTC. Setting condition: A 1-bit ECC error has occurred while the error flags DEDF and SEDF are 0.

(5) DFERSTC — Data Flash Error Status Clear Register

DFERSTC clears the error flags in the data flash error status register. DFERSTC is a write-only register and is always read as 0.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ERRCLR
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 29.20 DFERSTC Register Contents

Bit Position	Bit Name	Function
7 to 1	—	Reserved
0	ERRCLR	SEDF/DEDF Flag Clear Writing 1 to this bit clears the SEDF/DEDF flag.

(6) DFOVFSTR — Data Flash Error Overflow Status Register

DFOVFSTR monitors occurrence of data flash error overflow. The ERROVF flag is cleared by an internal reset, an external reset, or setting the ERROVFCLR bit to 1 in DFOVFSTC.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ERROVF
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 29.21 DFOVFSTR Register Contents

Bit Position	Bit Name	Function
7 to 1	—	Reserved
0	ERROVF	Error Overflow Flag This flag is set if an ECC error occurs while the error address register is full.

(7) DFOVFSTC — Data Flash Error Overflow Status Clear Register

DFOVFSTC clears the data flash error overflow flag. Setting the ERROVFCLR bit to 1 clears this flag.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ERROVFCLR
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 29.22 DFOVFSTC Register Contents

Bit Position	Bit Name	Function
7 to 1	—	Reserved
0	ERROVFCLR	Error Overflow Flag Clear Writing 1 to this bit clears the ERROVF flag. This bit is always read as 0.

(8) DFERRINT — Data Flash Error Notification Control Register

DFERRINT enables or disables generation of the error notification signal upon detection of a 2-bit ECC error or a 1-bit ECC error.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	DEDIE	SEDIE
Value after reset	0	0	0	0	0	0	1	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 29.23 DEFRRINT Register Contents

Bit Position	Bit Name	Function
7 to 2	—	Reserved
1	DEDIE	2-Bit ECC Error Notification Control Enables or disables generation of the error notification signal upon detection of a 2-bit error when ECC error detection and correction are enabled. 0: Disables notification of the 2-bit ECC error. 1: Enables notification of the 2-bit ECC error.
0	SEDIE	1-Bit ECC Error Notification Control Enables or disables generation of the error notification signal upon detection of a 1-bit error when ECC error detection and correction are enabled. 0: Disables notification of the 1-bit ECC error. 1: Enables notification of the 1-bit ECC error.

(9) DFEADR — Data Flash 1st Error Address Register

DFEADR holds the address at which an ECC error has occurred while both of the SEDF and DEDF bits in the data flash error status register are 0.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	DFEADR[20:16]				
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DFEADR[15:2]														—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 29.24 DFEADR Register Contents

Bit Position	Bit Name	Function
31 to 21	—	Reserved
20 to 2	DFEADR[20:2]	ECC Error Address These bits are read-only and used to monitor the address at which an ECC error has occurred.
1, 0	—	Reserved

(10) DFTSTCTL — Data Flash Test Control Register

DFTSTCTL is used for the ECC test. After ECC test mode is enabled by setting ECCTST = 1, the ECC bits can be read.

Set PROT[1:0] bits to 01_B when writing to this register.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PROT[1:0]		—	—	—	—	—	—	—	—	—	—	—	—	—	ECCTST
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 29.25 DFTSTCTL Register Contents

Bit Position	Bit Name	Function
15, 14	PROT[1:0]	These bits enable to write to this register. The read value is 0.
13 to 1	—	Reserved These bits are always read as 0. The write value should also be 0.
0	ECCTST	ECC Test Sets ECC test mode.

29.2.3.3 Test Function

Data in the ROM and the ECC bits can be read through the setting of the data flash test control register (DFTSTCTL).

(1) Reading the ROM Data

(1) Set the ECCDIS bit in the data flash ECC control register to 1 to disable ECC error detection and correction.

(2) When ECCDIS = 1, neither error detection nor correction proceeds when the data flash is read; the data output from the data flash is read unchanged.

How to exit this test mode:

(1) Set the ECCDIS bit in the data flash ECC control register to 0 to enable ECC error detection and correction.

(2) Reading the ECC Data

(1) Set the ECCDIS bit in the data flash ECC control register to 1 to disable ECC error detection and correction.

(2) Set the ECCTST bit in the data flash control register to 1 to set test mode.

(3) When the data flash is read, the 7 lower-order bits of read data are read as ECC data.

How to exit this test mode:

(1) Set the ECCDIS bit in the data flash ECC control register to 0 to enable ECC error detection and correction.

(2) Set the ECCTST bit in the data flash control register to 0 to set normal mode.

(3) Self-Diagnosis

Self-diagnosis of the ECC decoder is possible by writing incorrect data to the data flash memory beforehand (fault injection) and then reading this data. A 1- or 2-bit ECC error fault can be injected by generating correct ECC bits once and inverting only the appropriate bits.

For details on programming of the data flash, refer to "*RH850/E1x Flash Memory User's Manual: Hardware Interface*".

29.2.4 Local RAM (CPU1) ECC and Address Parity

29.2.4.1 Overview

The local RAM ECC of CPU1 is summarized in the table below.

Table 29.26 Overview of Local RAM ECC of CPU1

Item	Description
ECC error detection and correction	ECC error detection and correction can be either enabled or disabled. When enabled, either of the following settings can be selected. • ECC error detection and correction are carried out (2-bit error detection and 1-bit error detection and correction are carried out). • ECC error detection is carried out (2-bit error detection and 1-bit error detection are carried out). When disabled, neither error detection nor correction is carried out. In the state after reset, the ECC function is enabled; 1-bit error detection and correction and 2-bit error detection and notification are carried out.
Address parity	Address parity check can be either enabled or disabled. During data write, a parity bit generated based on the written address is written simultaneously with the write data. At this time, the same parity bit is written to two locations in the RAM. During read, comparison is performed between a parity bit generated based on the read address and a parity bit read from the RAM. The error decoding specifications are shown in **Table 29.27 Address Parity Definitions**. In the state after reset, this function is enabled.
Error notification	Upon occurrence of an ECC error or address parity error, the ECM is notified of the error. **ECC Error:** • Error notification can be either enabled or disabled upon detection of a 2-bit ECC error. • Error notification can be either enabled or disabled upon detection of a 1-bit ECC error. In the state after reset, notification of the 2-bit error is enabled and notification of the 1-bit error is disabled. **Parity Error:** • Error notification can be either enabled or disabled upon detection of an address parity error. • Error notification can be either enabled or disabled upon detection of a parity bit error. In the state after reset, notification of the address parity error is enabled and notification of the parity bit error is disabled. The error notification signal is output, where a 2-bit ECC error and an address parity error are handled as one source, and a 1-bit ECC error and a parity bit error are handled as one source.
Error status	A status register is provided, which indicates the statuses of 2-bit ECC error detection, 1-bit ECC error detection, and address parity error detection. If an error occurs while no error status flag is set, the corresponding status is set. The error status flag can be cleared using the clear register.
Self-diagnosis	Desired values can be written as RAM data and to the ECC and address parity bits. The RAM data and the ECC and address parity bits can be read directly.

The definitions of the address parity error during read accesses are shown in the table below.

Table 29.27 Address Parity Definitions

RAM Macro Parity Bit 1	RAM Macro Parity Bit 2	Read Address Parity	Error Determination and Error Name
0	0	0	No error
0	0	1	Address parity error
0	1	0	Parity bit error
0	1	1	Parity bit error
1	0	0	Parity bit error
1	0	1	Parity bit error
1	1	0	Address parity error
1	1	1	No error

CPU1 is capable of simultaneously writing or reading up to 128 bits of data at a time to or from the local RAM. Meanwhile, ECC and address parity bits are provided for each 32 bits of data and the locations for storage of each 32-bit data segment are referred to as banks 0 to 3. Addresses for the lowest-order bits (i.e., for the LSB side of the data) are for bank 0, while addresses for the highest-order bits (i.e., for the MSB side of the data) are for bank 3.

The relationship between addresses and banks is described below.

Table 29.28 Relationship between Addresses and Bank Numbers

Lower 4 Bits of Address (Hexadecimal Notation)	F_H to C_H	B_H to 8_H	7_H to 4_H	3_H to 0_H
Bank number	Bank 3	Bank 2	Bank 1	Bank 0

29.2.4.2 List of Registers

Table 29.29 List of Registers

Address	Symbol	Register Name	R/W	Value after Reset	Access Size
FFC6 5000$_H$	LRAPCTL_PE1	Local RAM address parity control register (PE1)	R/W	0000 0000$_H$	16/32
FFC6 5004$_H$	LRTSTCTL_PE1	Local RAM test control register	R/W	0000 0000$_H$	16/32
FFC6 5008$_H$	LRTDATBF0_PE1	Local RAM test data read buffer 0	R	0000 0000$_H$	8/16/32
FFC6 500C$_H$	LRTDATBF1_PE1	Local RAM test data read buffer 1	R	0000 0000$_H$	8/16/32
FFC6 5400$_H$	LRECCCTL_PE1	Local RAM ECC control register (PE1)	R/W	0000 0000$_H$	16/32
FFC6 5404$_H$	LRERRINT_PE1	Local RAM error information control register (PE1)	R/W	0000 0006$_H$	8/16/32
FFC6 5408$_H$	LRSTCLR_PE1	Local RAM status clear register (PE1)	W	0000 0000$_H$	8/16/32
FFC6 540C$_H$	LROVFSTR_PE1	Local RAM error count overflow status register (PE1)	R	0000 0000$_H$	8/16/32
FFC6 5410$_H$	LR1STERSTR_PE1	Local RAM 1st error status register (PE1)	R	0000 0000$_H$	8/16/32

29.2.4.3 Details of Registers

(1) LRAPCTL_PE1 — Local RAM Address Parity Control Register

LRAPCTL enables or disables address parity check. Set PROT[1:0] bits to 01_B when writing to this register.

This register is initialized by an internal reset or an external reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PROT[1:0]		—	—	—	—	—	—	—	—	—	—	—	—	—	APARIDIS
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 29.30 LRAPCTL Register Contents

Bit Position	Bit Name	Function
31 to 16	—	Reserved These bits are always read as 0. The write value should also be 0.
15, 14	PROT[1:0]	These bits enable to write to this register. The read value is 0.
13 to 1	—	Reserved These bits are always read as 0. The write value should also be 0.
0	APARIDIS	Address Parity Check Disable Enables or disables address parity check by all the address parity circuits (bank 0 to bank 3). 0: Enables address parity check. 1: Disables address parity check.

(2) LRTSTCTL_PE1 — Local RAM Test Control Register

This register is used for the ECC test (self-diagnosis) and address parity checker test (self-diagnosis). After ECC test mode is enabled by setting ECCTST = 1, desired values can be written to the ECC and address parity bits. The DATSEL bit is used to select RAM data or the ECC and address parity bits.

By setting address parity test mode (APTESTi = 1; i = 0, 1, 2, 3), the parity to be input to the address parity checker is inverted. Set PROT[1:0] bits to 01_B when writing to this register.

This register is initialized by an internal reset or an external reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PROT[1:0]		—	—	—	—	—	—	—	—	APTEST3	APTEST2	APTEST1	APTEST0	ECCTST	DATSEL
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Table 29.31 LRTSTCTL_PE1 Register Contents

Bit Position	Bit Name	Function
31 to 16	—	Reserved These bits are always read as 0. The write value should also be 0.
15, 14	PROT[1:0]	These bits enable to write to this register. The read value is 0.
13 to 6	—	Reserved These bits are always read as 0. The write value should also be 0.
5	APTEST3	Address Parity Checker (Bank 3) Test Sets the address parity checker to test mode. When APTEST3 = 1, the parity generated through the address parity generator is inverted.
4	APTEST2	Address Parity Checker (Bank 2) Test Sets the address parity checker to test mode. When APTEST2 = 1, the parity generated through the address parity generator is inverted.
3	APTEST1	Address Parity Checker (Bank 1) Test Sets the address parity checker to test mode. When APTEST1 = 1, the parity generated through the address parity generator is inverted.
2	APTEST0	Address Parity Checker (Bank 0) Test Sets the address parity checker to test mode. When APTEST0 = 1, the parity generated through the address parity generator is inverted.
1	ECCTST	ECC Test After ECC test mode is enabled by setting ECCTST = 1, the ECC and address parity bits can be read directly.
0	DATSEL	Data Select This bit is valid when ECCTST = 1. This bit selects the RAM bit which can be accessed when writing. 0: RAM data is selected. 1: The ECC bits and address parity bit are selected.

Note 1. When ECC test mode for the local RAM is enabled (ECCTST = 1), access to the local RAM should be accessed in 4-byte units.

(3) LRTDATBFn_PE1 — Local RAM Test Data Read Buffer n (n = 0, 1)

In ECC test mode (self-diagnosis), the ECC and address parity bits can be read. If the local RAM is read while ECCTST = 1 in the local RAM test control register LRTSTCTL, reading from the local RAM reads out the ECC and address parity bits, and these bits are stored in this buffer.

This register is initialized by an internal reset or an external reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	LRTDATBF								
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	LRTDATBF								
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 29.32 LRTDATBFn_PE1 Register Contents

Bit Position	Bit Name	Function
31 to 25	—	Reserved These bits are always read as 0. The write value should also be 0.
24 to 16	LRTDATBF	These bits are valid when ECCTST = 1 (selecting test mode) in the local RAM test control register. When reading from the corresponding bank in the local RAM, the ECC bits for the local RAM (bank (2n + 1)) and the address parity bit are respectively stored in LRTDATBF[22:16] and LRTDATBF[24:23].
15 to 9	—	Reserved These bits are always read as 0. The write value should also be 0.
8 to 0	LRTDATBF	These bits are valid when ECCTST = 1 (selecting test mode) in the local RAM test control register. When reading from the corresponding bank in the local RAM, the ECC bits for the local RAM (bank (2n)) and the address parity bit are respectively stored in LRTDATBF[6:0] and LRTDATBF[8:7].

(4) LRECCCTL_PE1 — Local RAM ECC Control Register

LRECCCTL enables or disables ECC error detection and correction and 1-bit error correction. Set PROT[1:0] bits to 01B when writing to this register.

This register is initialized by an internal reset or an external reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PROT[1:0]		—	—	—	—	—	—	—	—	—	—	—	—	SECDIS	ECCDIS
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Table 29.33 LRECCCTL_PE1 Register Contents

Bit Position	Bit Name	Function
31 to 16	—	Reserved These bits are always read as 0. The write value should also be 0.
15, 14	PROT[1:0]	These bits enable to write to this register. The read value is 0.
13 to 2	—	Reserved These bits are always read as 0. The write value should also be 0.
1	SECDIS	1-Bit Error Correction Disable Enables or disables 1-bit error correction when ECC error detection and correction are enabled. 0: Enables correction of the 1-bit error detected. 1: Disables correction of the 1-bit error detected.
0	ECCDIS	ECC Disable Enables or disables ECC error detection and correction. 0: Enables ECC error detection and correction. 1: Disables ECC error detection and correction.

(5) LRERRINT_PE1 — Local RAM Error Information Control

LRERRINT enables or disables generation of the error notification signal to the ECM upon detection of a 2-bit ECC error, a 1-bit ECC error, an address parity error, or a parity bit error.

LRERRINT is initialized by an internal reset or an external reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	PBEIE	APEIE	DEDIE	SEDIE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Table 29.34 LRERRINT_PE1 Register Contents

Bit Position	Bit Name	Function
31 to 4	—	Reserved These bits are always read as 0. The write value should also be 0.
3	PBEIE	Parity Bit Error Notification Enable Enables or disables generation of the error notification signal upon detection of a parity bit error when address parity check is enabled. 0: Disables notification of the parity bit error. 1: Enables notification of the parity bit error.
2	APEIE	Address Parity Error Notification Enable Enables or disables generation of the error notification signal upon detection of an address parity error when address parity check is enabled. 0: Disables notification of the address parity error. 1: Enables notification of the address parity error.
1	DEDIE	2-Bit ECC Error Notification Enable Enables or disables generation of the error notification signal upon detection of a 2-bit error when ECC error detection and correction are enabled. 0: Disables notification of the 2-bit ECC error. 1: Enables notification of the 2-bit ECC error.
0	SEDIE	1-Bit ECC Error Notification Enable Enables or disables generation of the error notification signal upon detection of a 1-bit error when ECC error detection and correction are enabled. 0: Disables notification of the 1-bit ECC error. 1: Enables notification of the 1-bit ECC error.

(6) LRSTCLR_PE1 — Local RAM Status Clear Register

LRSTCLR clears the error flags in the error status register (LR1STERSTR) and the overflow flag in the error count overflow status register (LROVFSTR). LRSTCLR is a write-only register and is always read as 0.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	STCLR3	STCLR2	STCLR1	STCLR0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	W	W	W	W

Table 29.35 LRSTCLR_PE1 Register Contents

Bit Position	Bit Name	Function
31 to 4	—	Reserved These bits are always read as 0. The write value should also be 0.
3	STCLR3	Error Overflow Flag Clear (for bank 3) Writing 1 to this bit clears the PBEF3, APEF3, DEDF3, and SEDF3 flags in LR1STERSTR; and ERROVF3 flag in LROVFSTR.
2	STCLR2	Error Overflow Flag Clear (for bank 2) Writing 1 to this bit clears the PBEF2, APEF2, DEDF2, and SEDF2 flags in LR1STERSTR; and ERROVF2 flag in LROVFSTR.
1	STCLR1	Error Overflow Flag Clear (for bank 1) Writing 1 to this bit clears the PBEF1, APEF1, DEDF1, and SEDF1 flags in LR1STERSTR; and ERROVF1 flag in LROVFSTR.
0	STCLR0	Error Overflow Flag Clear (for bank 0) Writing 1 to this bit clears the PBEF0, APEF0, DEDF0, and SEDF0 flags in LR1STERSTR; and ERROVF0 flag in LROVFSTR.

(7) LROVFSTR_PE1 — Local RAM Error Count Overflow Status Register

LROVFSTR monitors occurrence of error overflow. If the second error occurs after the first error (= while any of the error flags in the error status register is set), the flag in this register is set. However, if the second error is identical to the first error (both the source and address are same), this flag is not set. ERROVF is cleared by an internal reset, an external reset, or setting the STCLR bit to 1 in LRSTCLR.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	ERROVF3	ERROVF2	ERROVF1	ERROVF0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 29.36 LROVESTR_PE1 Register Contents

Bit Position	Bit Name	Function
31 to 4	—	Reserved These bits are always read as 0. The write value should also be 0.
3	ERROVF3	Error Overflow Flag (for bank 3) This flag is set if the second error occurs while any of the error flags (PBEF3, APEF3, DEDF3, and SEDF3) in the error status register is set, except when both of the error address and source of the second error are the same as those of the first error.
2	ERROVF2	Error Overflow Flag (for bank 2) This flag is set if the second error occurs while any of the error flags (PBEF2, APEF2, DEDF2, and SEDF2) in the error status register is set, except when both of the error address and source of the second error are the same as those of the first error.
1	ERROVF1	Error Overflow Flag (for bank 1) This flag is set if the second error occurs while any of the error flags (PBEF1, APEF1, DEDF1, and SEDF1) in the error status register is set, except when both of the error address and source of the second error are the same as those of the first error.
0	ERROVF0	Error Overflow Flag (for bank 0) This flag is set if the second error occurs while any of the error flags (PBEF0, APEF0, DEDF0, and SEDF0) in the error status register is set, except when both of the error address and source of the second error are the same as those of the first error.

(8) LR1STERSTR_PE1 — Local RAM 1st Error Status Register

LR1STERSTR monitors occurrence of the first error. Errors are detected and the states are updated independently for each bank.

An error status flag is set if an error occurs while all error flags for the given bank are 0. Note, however, that the corresponding error flag is also set if a 2-bit ECC error or an address parity error occurs in subsequent access as long as only the 1-bit ECC error flag or parity bit error flag is set at the time.

If more than one error is detected for a given memory access, all the corresponding error flags are set. LR1STERSTR is cleared by an internal reset, an external reset, or setting 1 to the STCLR bit in LRSTCLR.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	PBEF3	APEF3	DEDF3	SEDF3	—	—	—	—	PBEF2	APEF2	DEDF2	SEDF2
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	PBEF1	APEF1	DEDF1	SEDF1	—	—	—	—	PBEF0	APEF0	DEDF0	SEDF0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 29.37 LR1STERSTR_PE1 Register Contents

Bit Position	Bit Name	Function
7+8n:4+8n	—	Reserved These bits are always read as 0. The write value should also be 0.
3+8n	PBEFn	Parity Bit Error Monitor Flag Clearing condition: Cleared to 0 by a reset or by setting 1 to the STCLRn bit in LRSTCLR. Setting condition: A parity bit error has occurred while all the error flags PBEFn, APEFn, DEDFn, and SEDFn are 0.
2+8n	APEFn	Address Parity Error Monitor Flag Clearing condition: Cleared to 0 by a reset or by setting 1 to the STCLRn bit in LRSTCLR. Setting condition: An address parity error has occurred while the error flags DEDFn and APEFn are 0. **NOTE** This flag is set equally for read and write; the error generation source is not considered.
1+8n	DEDFn	2-Bit ECC Error Monitor Flag Clearing condition: Cleared to 0 by a reset or by setting 1 to the STCLRn bit in LRSTCLR. Setting condition: A 2-bit ECC error has occurred while the error flags DEDFn and APEFn are 0.
0+8n	SEDFn	1-Bit ECC Error Monitor Flag Clearing condition: Cleared to 0 by a reset or by setting 1 to the STCLRn bit in LRSTCLR. Setting condition: A 1-bit ECC error has occurred while all the error flags PBEFn, APEFn, DEDFn, and SEDF are 0.

Note 1. n = 0 to 3, where “n” denotes a bank number.

29.2.4.4 Test Function

Through appropriate register setting, desired values can be written as RAM data and to the ECC and address parity bits. Also, data in the RAM and the ECC and address parity bits can all be read.

(1) Writing RAM Data

(a) Set the ECCTST bit in the local RAM test control register to 1 to set test mode.

(b) Set the corresponding DATSEL bit in the local RAM test control register to 0 to select RAM data for access when writing.

(c) When data is written to the local RAM, only the RAM data can be modified without updating the ECC bits.

How to exit this test mode:

(a) Set the ECCTST bit in the local RAM test control register to 0 to disable test mode (normal mode).

(2) Reading RAM Data

(a) Set the ECCDIS bit in the local RAM ECC control register to 1 to disable ECC error detection and correction.

(b) Read the local RAM. Since neither error detection nor correction proceeds when the local RAM is read, the RAM data is read unchanged.

How to exit this test mode:

(a) Set the ECCDIS bit in the local RAM ECC control register to 0 to enable ECC error detection and correction.

(3) Writing to the ECC and Address Parity Bits

(a) Set the ECCTST bit in the local RAM test control register to 1 to set test mode.

(b) Set the corresponding DATSEL bit in the local RAM test control register to 1 to select the ECC bits and the address parity bit for access when writing.

(c) When data is written to the local RAM, only the ECC and address parity bits can be modified without updating the RAM data. At that time, bit[6:0] and bit[8:7] are respectively written to the ECC bits and to the address parity bit.

NOTE

Writing desired values to the RAM data and the address parity bit should be in order of 1) writing the RAM data and then 2) writing the address parity bit.
When writing desired values to the RAM data and ECC bits, you can start either by writing to the RAM data or ECC bits.

How to exit this test mode:

(a) Set the ECCTST bit in the local RAM test control register to 0 to disable test mode (normal mode).

(4) Reading the ECC and Address Parity Bits

(a) Set the ECCTST bit in the local RAM test control register to 1 to set test mode.

(b) When the local RAM is read, the ECC and address parity bits are stored in the bank corresponding to local RAM test data read buffer 0 or local RAM test data read buffer 1.

How to exit this test mode:

(a) Set the ECCTST bit in the local RAM test control register to 0 to disable test mode (normal mode).

(5) Self-Diagnosis of the ECC Check Function

Desired values can be written to the RAM data and/or ECC bits by following the procedure described in (1) or (3) above. Therefore, a fault can be injected by, for example, inverting appropriate bits of the RAM data and/or ECC bits. After that, self-diagnosis of the ECC decoder is possible by reading the local RAM in normal mode and checking the result of error correction or detection.

(6) Self-Diagnosis of the Address Parity Check Function

Self-diagnosis is enabled by following either of the two procedures below.

(a) Setting APTESTi (i = 0, 1, 2, 3) in the local RAM test control register to 1 inverts the result of address parity generation for the corresponding bank. That is, a fault can be injected to the address parity generator. Writing to the corresponding bank in the local RAM in this state and checking the result of parity error detection allows self-diagnosis of the address parity check function in writing.

(b) Desired data can be written to the address parity bit through the procedure described in (3) above. This enables injection of a 1-bit or 2-bit fault to the address parity bit by inverting the address parity bit. After that, self-diagnosis of the address parity bit checking function for reading is possible by reading the local RAM in normal mode and checking the result of parity error detection.

29.2.5 Local RAM (PCU) ECC

29.2.5.1 Overview

The local RAM ECC of PCU is summarized in the table below.

Table 29.38 Overview of Local RAM ECC of PCU

Item	Description
ECC error detection and correction	ECC error detection and correction can be either enabled or disabled. When enabled, either of the following settings can be selected. • ECC error detection and correction are carried out (2-bit error detection and 1-bit error detection and correction are carried out). • ECC error detection is carried out (2-bit error detection and 1-bit error detection are carried out). When disabled, neither error detection nor correction is carried out. In the state after reset, the ECC function is enabled; 1-bit error detection and correction and 2-bit error detection and notification are carried out.
Address parity	None
Error notification	Upon occurrence of an ECC error, the ECM is notified of the error. **ECC Error:** • Error notification can be either enabled or disabled upon detection of a 2-bit ECC error. • Error notification can be either enabled or disabled upon detection of a 1-bit ECC error. In the state after reset, notification of 2-bit error is enabled and notification of 1-bit error is disabled. The error notification signal is output, where a 2-bit ECC error is handled as one source and a 1-bit ECC error is handled as one source.
Error status	A status register is provided, which indicates the statuses of 2-bit ECC error detection and 1-bit ECC error detection. If an error occurs while no error status flag is set, the corresponding status is set. The error status flag can be cleared using the clear register.
Address capture	If an ECC error occurs while no error status flag is set, the address at which the associated error has occurred is captured. The address is captured when a 2-bit ECC error or a 1-bit ECC error is detected. The error status flag serves as the enable bit of the capture address.
Self-diagnosis	Desired values can be written as RAM data and to the ECC bits. Data in the RAM and the ECC bits can be read directly.

29.2.5.2 List of Registers

Table 29.39 List of Registers

Address	Symbol	Register Name	R/W	Value after Reset	Access Size
FFC6 5800$_H$	LRECCCTL_PCU	Local RAM ECC control register (PCU)	R/W	0000 0000$_H$	16/32
FFC6 5804$_H$	LRFSTERSTR_PCU	Local RAM 1st error status register (PCU)	R	0000 0000$_H$	32
FFC6 5824$_H$	LRSTCLR_PCU	Local RAM error status clear register (PCU)	W	0000 0000$_H$	8/16/32
FFC6 5828$_H$	LROVFSTR_PCU	Local RAM overflow status register (PCU)	R	0000 0000$_H$	32
FFC6 582C$_H$	LROVFSTC_PCU	Local RAM overflow status clear register (PCU)	W	0000 0000$_H$	8/16/32
FFC6 5830$_H$	LRFSTEADR0_PCU	Local RAM 1st error address register 0 (PCU)	R	0000 0000$_H$	32
FFC6 58B0$_H$	LRERRINT_PCU	Local RAM error information control register (PCU)	R/W	0000 0002$_H$	8/16/32
FFC6 58B4$_H$	LRTSTCTL_PCU	Local RAM test control register (PCU)	R/W	0000 0000$_H$	16/32
FFC6 58B8$_H$	LRTDATBF0_PCU	Local RAM test data read buffer (PCU)	R	0000 0000$_H$	32

29.2.5.3 Details of Registers

(1) LRECCCTL_PCU — Local RAM ECC Control Register

LRECCCTL enables or disables ECC error detection and correction and 1-bit error correction. Set PROT[1:0] bits to 01B when writing to this register.

This register is initialized by an internal reset or an external reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PROT[1:0]		—	—	—	—	—	—	—	—	—	—	—	—	SECDIS	ECCDIS
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Table 29.40 LRECCCTL_PCU Register Contents

Bit Position	Bit Name	Function
31 to 16	—	Reserved These bits are always read as 0. The write value should also be 0.
15, 14	PROT[1:0]	These bits enable to write to this register. The read value is 0.
13 to 2	—	Reserved These bits are always read as 0. The write value should also be 0.
1	SECDIS	1-Bit Error Correction Disable Enables or disables 1-bit error correction when ECC error detection and correction are enabled. 0: Enables correction of the 1-bit error detected. 1: Disables correction of the 1-bit error detected.
0	ECCDIS	ECC Disable Enables or disables ECC error detection and correction. 0: Enables ECC error detection and correction. 1: Disables ECC error detection and correction.

(2) LRFSTERSTR_PCU — Local RAM 1st Error Status Register

LRFSTERSTR monitors occurrence of the first error.

An error status bit is set if an error occurs while the error flag is 0. Note, however, that the corresponding error flag is also set if a 2-bit ECC error occurs in subsequent access as long as only the 1-bit ECC error flag is set at the time.

LRFSTERSTR is cleared by an internal reset or an external reset, or by setting the FSTERRCLR0 bit in LRSTCLR to 1.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DEDF0	SEDF0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 29.41 LRFSTERSTR_PCU Register Contents

Bit Position	Bit Name	Function
31 to 2	—	Reserved These bits are always read as 0. The write value should also be 0.
1	DEDF0	2-Bit ECC Error Monitor Flag Clearing condition: Cleared to 0 by a reset or by setting 1 to the FSTERRCLR0 bit in LRSTCLR. Setting condition: A 2-bit ECC error has occurred while the error flag DEDF0 is 0.
0	SEDF0	1-Bit ECC Error Monitor Flag Clearing condition: Cleared to 0 by a reset or by setting 1 to the FSTERRCLR0 bit in LRSTCLR. Setting condition: A 1-bit ECC error has occurred while the error flags DEDF0 and SEDF0 are 0.

(3) LRSTCLR_PCU — Local RAM Error Status Clear Register

LRSTCLR clears the error flags in the local RAM 1st error status register (LRFSTERSTR). LRSTCLR is a write-only register and is always read as 0.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	FSTERRCLR0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	W

Table 29.42 LRSTCLR_PCU Register Contents

Bit Position	Bit Name	Function
31 to 1	—	Reserved These bits are always read as 0. The write value should also be 0.
0	FSTERRCLR0	1st Error Status Register Clear Writing 1 to this bit clears the error flags in the 1st error status register.

(4) LROVFSTR_PCU — Local RAM Overflow Status Register

LROVFSTR monitors occurrence of local RAM error overflow. ERROVF0 is cleared by an internal reset, an external reset, or setting the ERROVFCLR0 bit to 1 in LROVFSTC.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ERROVF0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 29.43 LROVFSTR_PCU Register Contents

Bit Position	Bit Name	Function
31 to 1	—	Reserved These bits are always read as 0. The write value should also be 0.
0	ERROVF0	Error Overflow Flag This flag is set if the second error occurs while any of the error flags (DEDF0 and SEDF0) in the error status register is set, except when both of the error address and source of the second error are the same as those of the first error.

(5) LROVFSTC_PCU — Local RAM Overflow Status Clear Register

LROVFSTC clears the local RAM error overflow flag. Setting the ERROVFCLR0 bit to 1 clears the flag.

LROVFSTC is initialized by an internal reset or an external reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ERROVFCLR0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	W

Table 29.44 LROVFSTC_PCU Register Contents

Bit Position	Bit Name	Function
31 to 1	—	Reserved These bits are always read as 0. The write value should also be 0.
0	ERROVFCLR0	Error Overflow Flag Clear Writing 1 to this bit clears the ERROVF0 flag. This bit is always read as 0.

(6) LRFSTEADR0_PCU — Local RAM 1st Error Address Register 0

LRFSTEADR0 holds the address at which an ECC error has occurred. LRFSTEADR0[21:2] of this register correspond to bits [21:2] of the real address. The real address can be calculated by appending the higher-order address bits [31:22] as a base address.

LRFSTEADR0 is initialized by an internal reset or an external reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	LRFSTEADR0[21:16]					
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	LRFSTEADR0[15:2]														—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 29.45 LRFSTEADR0_PCU Register Contents

Bit Position	Bit Name	Function
31 to 22	—	Reserved These bits are always read as 0. The write value should also be 0.
21 to 2	LRFSTEADR0 [21:2]	1st Error Address These bits are read-only and used to monitor the address of the first error. The address is updated under the following conditions: • When error source for the retained address is SEDF A 1-bit error has occurred on the same address -> The address is not overwritten. A 1-bit error has occurred on a different address -> The address is not overwritten. A 2-bit error has occurred -> The address is overwritten. • When error source for the retained address is DEDF A 1-bit error has occurred -> The address is not overwritten. A 2-bit error has occurred on the same address -> The address is not overwritten. A 2-bit error has occurred on a different address -> The address is not overwritten.
1, 0	—	Reserved These bits are always read as 0. The write value should also be 0.

(7) LRERRINT_PCU — Local RAM Error Information Control Register

LRERRINT enables or disables generation of the error notification signal to the ECM upon detection of a 2-bit ECC error or a 1-bit ECC error.

LRERRINT is initialized by an internal reset or an external reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DEDIE	SEDIE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Table 29.46 LRERRINT_PCU Register Contents

Bit Position	Bit Name	Function
31 to 2	—	Reserved These bits are always read as 0. The write value should also be 0.
1	DEDIE	2-Bit ECC Error Notification Enable Enables or disables generation of the error notification signal upon detection of a 2-bit error when ECC error detection and correction are enabled. 0: Disables notification of the 2-bit ECC error. 1: Enables notification of the 2-bit ECC error.
0	SEDIE	1-Bit ECC Error Notification Enable Enables or disables generation of the error notification signal upon detection of a 1-bit error when ECC error detection and correction are enabled. 0: Disables notification of the 1-bit ECC error. 1: Enables notification of the 1-bit ECC error

(8) LRTSTCTL_PCU — Local RAM Test Control Register

This register is used for the ECC test (self-diagnosis). After ECC test mode is enabled by setting ECCTST = 1, desired values can be written to the ECC bits. The DATSEL bit is used to select the RAM data or ECC bits. Set PROT[1:0] bits to 01_B when writing to this register.

This register is initialized by an internal reset or an external reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PROT[1:0]		—	—	—	—	—	—	—	—	—	—	—	—	ECCTST	DATSEL
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Table 29.47 LRTSTCTRL_PCU Register Contents

Bit Position	Bit Name	Function
31 to 16	—	Reserved These bits are always read as 0. The write value should also be 0.
15, 14	PROT[1:0]	These bits enable to write to this register. The read value is 0.
13 to 2	—	Reserved These bits are always read as 0. The write value should also be 0.
1	ECCTST	ECC Test After ECC test mode is enabled by setting ECCTST = 1, the ECC bits can be read directly.
0	DATSEL	Data Select This bit is valid when ECCTST = 1. This bit selects the RAM bit which can be accessed when writing. 0: RAM data is selected. 1: The ECC bits are selected.

Note 1. When ECC test mode for the local RAM is enabled (ECCTST = 1), access to the local RAM should be accessed in 4-byte units.

(9) LRTDATBF0_PCU — Local RAM Test Data Read Buffer 0

In ECC test mode (self-diagnosis), the ECC bits can be read. If the local RAM is read while ECCTST = 1 in the local RAM test control register LRTSTCTL, reading from the local RAM reads out the ECC bits, and these bits are stored in this buffer.

This register is initialized by an internal reset or an external reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	LRTDATBF						
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 29.48 LRTDATBF0_PCU Register Contents

Bit Position	Bit Name	Function
31 to 7	—	Reserved These bits are always read as 0. The write value should also be 0.
6 to 0	LRTDATBF	These bits are valid when ECCTST = 1 (selecting test mode) in the local RAM test control register. When reading from the local RAM, the ECC bits are stored in LRTDATBF[6:0].

29.2.5.4 Test Function

For the test function and procedures for self-diagnosis, see the section on the local RAM (for the CPU). However, since the local RAM for the PCU does not have an address parity facility, ignore the description of address parity.

29.2.6 Global RAM ECC and Address Parity

29.2.6.1 Overview

The global RAM ECC is summarized in the table below.

Table 29.49 Overview of Global RAM ECC

Item	Description
ECC error detection and correction	ECC error detection and correction can be either enabled or disabled. When enabled, either of the following settings can be selected. • ECC error detection and correction are carried out (2-bit error detection and 1-bit error detection and correction are carried out). • ECC error detection is carried out (2-bit error detection and 1-bit error detection are carried out). When disabled, neither error detection nor correction is carried out. In the state after reset, the ECC function is enabled, and 1-bit error detection and correction and 2-bit error detection and notification are carried out.
Address parity	Address parity check can be either enabled or disabled. During data write, a parity bit generated based on the written address is written simultaneously with the write data. At this time, the parity bit is written to only one location in the RAM. During data read, comparison is performed between a parity bit generated based on the read address and a parity bit read from the memory. In the state after reset, this function is enabled.
Error notification	Upon occurrence of an ECC error or address parity error, the ECM is notified of the error. **ECC Error:** • Error notification can be either enabled or disabled upon detection of a 2-bit ECC error. • Error notification can be either enabled or disabled upon detection of a 1-bit ECC error. In the state after reset, error notification is enabled upon detection of a 2-bit ECC error, and error notification is disabled upon detection of a 1-bit ECC error. Parity Error: • Error notification can be either enabled or disabled upon detection of an address parity error. In the state after reset, error notification is enabled upon detection of an address parity error. The error notification signal is output, where a 2-bit ECC error and an address parity error are handled as one source, and a 1-bit ECC error is handled as one source.
Error status	A status register is provided, which indicates the statuses of 2-bit ECC error detection, 1-bit ECC error detection, and address parity error detection. If an error occurs while no error status flag is set, the corresponding status is set. The error status flag can be cleared using the clear register.
Address capture	If an ECC error or a parity error occurs while no error status flag is set, the address at which the associated error has occurred is captured. The address is captured when a 2-bit ECC error, a 1-bit ECC error, or an address parity error is detected. The error status flag serves as the enable bit of the capture address.
Self-diagnosis	Desired values can be written as RAM data and to the ECC and address parity bits. Data in the RAM and the ECC and address parity bits can be read directly.

The ECC encoder and decoder and the address parity generator are provided for the access ports (CPU1 and interconnect) that are connected to the global RAM. Separate address checkers are provided for bank A and bank B of the global RAM. Also, the ECC decoder and ECC encoder for RMW processing* are provided for each of bank A and bank B. See **Figure 29.2**.

- RMW processing

  Bit manipulation instructions, 2-byte writing and 1-byte writing are executed in three steps: 1) reading of 32-bit data; 2) generation of write data by replacement of (modifying) the predetermined data; and 3) writing of 32-bit data. In this section, such operations are referred to as read-modify-write (RMW) processing. RMW processing for the global RAM proceeds in the controller for each bank.

  In RWM processing, the ECC is decoded for the read operation in step 1), while the ECC is encoded for the write operation in step 3).

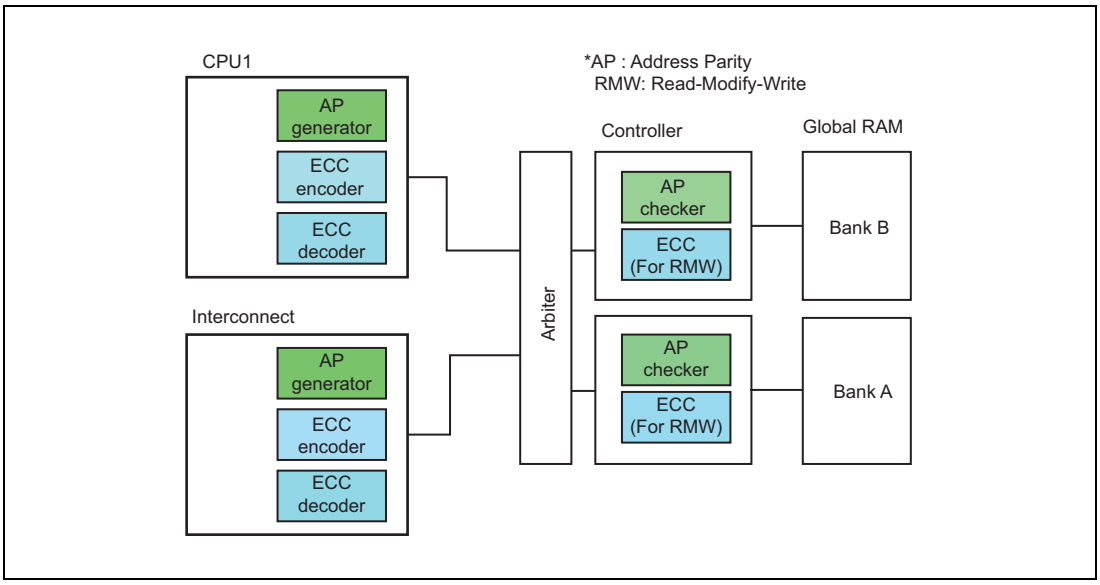


Figure 29.2 ECC of Global RAM and Address Parity

Up to 64 bits of data can be simultaneously read from or written to the global RAM. Meanwhile, the ECC and address parity are provided for each 32-bit data. That is, two each of the ECC decoder, ECC encoder, ECC circuit (for RMW), and address parity checker in **Figure 29.2** are provided, one for the 32 higher-order bits and the other for the 32 lower-order bits.

Table 29.50 Address and Corresponding ECC Circuit

Lower 3 Bits of the Address	7_H to 4_H	3_H to 0_H
Corresponding ECC circuit	Upper 32 bits	Lower 32 bits

NOTE

In the local RAM (CPU1), the locations for storage of data for units of the ECC are referred to as banks 0 to 3. Since the global RAM is divided into banks A and B with the address FEF0 0000_H as the boundary, banks A and B are respectively referred to as the 32 higher-order bits and the 32 lower-order bits to reduce confusion.

29.2.6.2 List of Registers

Table 29.51 List of Registers

Address	Symbol*[1]	Register Name	R/W	Value after Reset	Access Size
FFC6 4000$_H$	GRECCCTL_GRAMC	Global RAM ECC control register (GRAMC)	R/W	0000 0000$_H$	16/32
FFC6 4004$_H$	GRTSTCTL	Global RAM test control register	R/W	0000 0000$_H$	16/32
FFC6 4008$_H$	GRTDATBF0	Global RAM test data read buffer 0	R	0000 0000$_H$	8/16/32
FFC6 400C$_H$	GRTDATBF1	Global RAM test data read buffer 1	R	0000 0000$_H$	8/16/32
FFC6 4010$_H$	GRTDATBF2	Global RAM test data read buffer 2	R	0000 0000$_H$	8/16/32
FFC6 4014$_H$	GRTDATBF3	Global RAM test data read buffer 3	R	0000 0000$_H$	8/16/32
FFC6 4018$_H$	GRDECINBF0	Global RAM ECC decoder input data buffer 0	R/W	0000 0000$_H$	8/16/32
FFC6 401C$_H$	GRDECINBF1	Global RAM ECC decoder input data buffer 1	R/W	0000 0000$_H$	8/16/32
FFC6 4200$_H$	GRECCCTL_VCI	Global RAM ECC control register (VCI)	R/W	0000 0000$_H$	16/32
FFC6 4204$_H$	GRERRINT_VCI	Global RAM error information control register (VCI)	R/W	0000 0006$_H$	8/16/32
FFC6 4208$_H$	GRSTCLR_VCI	Global RAM status clear register (VCI)	W	0000 0000$_H$	8/16/32
FFC6 420C$_H$	GROVFSTR_VCI	Global RAM error count overflow status register (VCI)	R	0000 0000$_H$	8/16/32
FFC6 4210$_H$	GR1STERSTR_VCI	Global RAM 1st error status register (VCI)	R	0000 0000$_H$	8/16/32
FFC6 4250$_H$	GR1STEADR0_VCI	Global RAM 1st error (lower 32-bit data) address register (VCI)	R	0000 0000$_H$	8/16/32
FFC6 4254	GR1STEADR1_VCI	Global RAM 1st error (upper 32-bit data) address register (VCI)	R	0000 0000$_H$	8/16/32
FFC6 4400$_H$	GRECCCTL_PE1	Global RAM ECC control register (PE1)	R/W	0000 0000$_H$	16/32
FFC6 4404$_H$	GRERRINT_PE1	Global RAM error information control register (PE1)	R/W	0000 0006$_H$	8/16/32
FFC6 4408$_H$	GRSTCLR_PE1	Global RAM status clear register (PE1)	W	0000 0000$_H$	8/16/32
FFC6 440C$_H$	GROVFSTR_PE1	Global RAM error count overflow status register (PE1)	R	0000 0000$_H$	8/16/32
FFC6 4410$_H$	GR1STERSTR_PE1	Global RAM 1st error status register (PE1)	R	0000 0000$_H$	8/16/32
FFC6 4450$_H$	GR1STEADR0_PE1	Global RAM 1st error (lower 32-bit data) address register (PE1)	R	0000 0000$_H$	8/16/32
FFC6 4454$_H$	GR1STEADR1_PE1	Global RAM 1st error (upper 32-bit data) address register (PE1)	R	0000 0000$_H$	8/16/32

Note 1. In the register symbols, "*_VCI" and "*_PE1" indicate the registers provided for the ECC controllers for each of the access ports. Specifically, "_VCI" and "_PE1" indicate the control registers for access to the Global RAM from system interconnect 1 and to the Global RAM from CPU1, respectively. "_GRAMC" is a common control register for all access ports.

29.2.6.3 Details of Registers

(1) GRECCCTL_GRAMC — Global RAM ECC Control Register

GRECCCTL is the ECC and address parity control register shared by Global RAMs. GRECCCTL enables or disables address parity check and specifies ECC processing for Read-Modify-Write (RMW) processing. Set PROT[1:0] bits to 01_B when writing to this register.

This register is initialized by an internal reset or an external reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PROT[1:0]		—	—	—	—	—	—	—	—	—	—	—	APARIDIS	SECDIS	ECCDIS
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Table 29.52 GRECCCTL_GRAMC Register Contents

Bit Position	Bit Name	Function
31 to 16	—	Reserved These bits are always read as 0. The write value should also be 0.
15, 14	PROT[1:0]	These bits enable to write to this register. The read value is 0.
13 to 3	—	Reserved These bits are always read as 0. The write value should also be 0.
2	APARIDIS	Address Parity Check Disable Enables or disables address parity check. In the initial state, the parity check is enabled. 0: Enables address parity check. 1: Disables address parity check.
1	SECDIS	1-Bit Error Correction Disable for RMW Enables or disables 1-bit error correction when ECC error detection and correction are enabled. 0: Enables correction of the 1-bit error detected. 1: Disables correction of the 1-bit error detected.
0	ECCDIS	ECC Disable for RMW Enables or disables ECC error detection and correction. In the initial state, ECC error detection and correction are enabled. 0: Enables ECC error detection and correction. 1: Disables ECC error detection and correction. **NOTE** The encoding function is enabled even though the error detection and correction are disabled.

(2) GRTSTCTL — Global RAM Test Control Register

This register is used for the ECC test (self-diagnosis) and address parity checker test (self-diagnosis). After ECC test mode is enabled by setting ECCTST = 1, desired values can be written to the ECC and address parity bits. The DATSEL0 or DATSEL1 bit is used to select the RAM data or the ECC and address parity bits. Also, input and output by the ECC decoder in the Global RAM controller can be controlled for testing (self-diagnosis).

By setting address parity test mode (APTESTi = 1; i = 0, 1, 2, 3), the parity to be input to the address parity checker is inverted. Set PROT[1:0] bits to 01_B when writing to this register.

This register is initialized by an internal reset or an external reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PROT[1:0]		—	—	—	—	—	—	APTEST3	APTEST2	APTEST1	APTEST0	ECCTST	DECINEN	DATSEL	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 29.53 GRTSTCTL Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 16	—	Reserved These bits are always read as 0. The write value should also be 0.
15, 14	PROT[1:0]	These bits enable to write to this register. The read value is 0.
13 to 8	—	Reserved These bits are always read as 0. The write value should also be 0.
7	APTEST3	Address Parity Checker (Bank B, 32 Higher-Order Bits) Test Sets the address parity checker to test mode. When APTEST3 = 1, the parity generated through the address parity generator is inverted.
6	APTEST2	Address Parity Checker (Bank B, 32 Lower-Order Bits) Test Sets the address parity checker to test mode. When APTEST2 = 1, the parity generated through the address parity generator is inverted.
5	APTEST1	Address Parity Checker (Bank A, 32 Higher-Order Bits) Test Sets the address parity checker to test mode. When APTEST1 = 1, the parity generated through the address parity generator is inverted.
4	APTEST0	Address Parity Checker (Bank A, 32 Lower-Order Bits) Test Sets the address parity checker to test mode. When APTEST0 = 1, the parity generated through the address parity generator is inverted.
3	ECCTST	ECC Test After ECC test mode is enabled by setting ECCTST = 1, the ECC and address parity bits can be read directly.
2	DECINEN	RMW ECC Decoder Error Injection Enable This bit is valid when ECCTST = 1. This bit enables input of the value in the ECC decoder input buffer register to the ECC decoder for use in updating of data at the time of RMW access. 0: The value in the ECC decoder input buffer is not input. 1: The value in the ECC decoder input buffer is input.

Table 29.53 GRTSTCTL Register Contents (2/2)

Bit Position	Bit Name	Function
1, 0	DATSEL	Read Buffer Storage Data Select 0 and 1 These bits are valid when ECCTST = 1. These bits select the value to be stored in the read buffer GRTDATBFn, and the value to be written to each field. 00: – GRTDATBFn: When reading involves an RMW cycle, the ECC and address parity bits are stored. – Global RAM: When writing involves an RMW cycle, the data area that is updated depends on the unit of access and the location accessed. The ECC bits are not updated. 01: – GRTDATBFn: When reading involves an RMW cycle, the ECC and address parity bits are stored. – Global RAM: When writing involves an RMW cycle, only the ECC and address parity bits are updated. The data area is not updated. 10: – GRTDATBFn: When access is RMW, this register holds the result of ECC decoding for the data read out in the read portion of the RMW cycle. Its value is not updated in the case of access that is not RMW. – Global RAM: Operates in the same way as in normal operating mode (i.e. when ECCTST = 0). 11: – GRTDATBFn: When access is RMW, this register holds the result of ECC decoding for use in the updating of data at the time of RMW access. Its value is not updated in the case of access that is not RMW. – Global RAM: Operates in the same way as in normal operating mode (i.e. when ECCTST = 0). In any case, the result of reading by the CPU, DMAC, etc. is the same value as would normally be read out.

(3) GRTDATBFn — Global RAM Test Data Read Buffer n (n = 0 to 3)

In test mode (ECCTST = 1), data in the RAM, the ECC bits, address parity bit, and ECC decoder output can all be read. When the RAM is read, the value selected by the DATSEL1 or DATSEL0 bit of the global RAM test control register is stored in this buffer.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GRTDATABF															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GRTDATABF															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 29.54 GRTDATBFn Register Contents

Bit Position	Bit Name	Function
31 to 0	GRTDATBF	These bits are valid when while ECCTST = 1 (selecting test mode) in the global RAM test control register. When (DATSEL1, DATSEL0) = (0, 0) or (0, 1) When reading from the RAM, the ECC bits and the address parity bit are respectively stored in GRTDATBF[6:0] and GRTDATBF[7]. 0 is stored in GRTDATBF[31:8]. When (DATSEL1, DATSEL0) = (1, 0) When access is RMW, the output data from the ECC decoder for reading (after updating) are stored in GRTDATBF[31:0]. When (DATSEL1, DATSEL0) = (1, 1) When access is RMW, the output data from the ECC decoder for updating (after updating) are stored in GRTDATBF[31:0].

n = 0: Bank A, 32 lower-order bits
n = 1: Bank A, 32 higher-order bits
n = 2: Bank B, 32 lower-order bits
n = 3: Bank B, 32 higher-order bits

(4) GRECCCTL_VCI/_PE1 — Global RAM ECC Control Register

GRECCCTL enables or disables ECC error detection and correction and 1-bit error correction. Set PROT[1:0] bits to 01_B when writing to this register.

The setting of this register is used for accesses through the respective access port.

This register is initialized by an internal reset or an external reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PROT[1:0]		—	—	—	—	—	—	—	—	—	—	—	—	SECDIS	ECCDIS
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Table 29.55 GRECCCTL_VCI/PE1 Register Contents

Bit Position	Bit Name	Function
13 to 16	—	Reserved These bits are always read as 0. The write value should also be 0.
15, 14	PROT[1:0]	These bits enable to write to this register. The read value is 0.
13 to 2	—	Reserved These bits are always read as 0. The write value should also be 0.
1	SECDIS	1-Bit Error Correction Disable Enables or disables 1-bit error correction when ECC error detection and correction are enabled. 0: Enables correction of the 1-bit error detected. 1: Disables correction of the 1-bit error detected.
0	ECCDIS	ECC Disable Enables or disables ECC error detection and correction. 0: Enables ECC error detection and correction. 1: Disables ECC error detection and correction.

(5) GRDECINBF0 — Global RAM ECC Decoder Input Data Buffer 0

This register holds input data for the ECC decoder to use in updating the data at the time of RMW access. The value of this register is treated by the ECC decoder as if it were 32 bits of data from RAM.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	GRDECINBF0															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GRDECINBF0															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 29.56 GRDECINBF0 Register Contents

Bit Position	Bit Name	Function
31 to 0	GRDECINBF0	These bits are valid when ECCTST = 1 (selecting test mode) in the global RAM test control register. When DECINEN = 1, the value of this register is input to the ECC decoder as data for use in updating in response to the execution of an RMW instruction. The value is treated as if it were 32 bits of data from RAM. This register is common to banks A and B and provides the values for both the 32 higher-order bits and the 32 lower-order bits.

(6) GRDECINBF1 — Global RAM ECC Decoder Input Data Buffer 1

This register holds input data for the ECC decoder to use in updating the data at the time of RMW access. The value of this register is treated by the ECC decoder as if it were 7 bits of data from ECC.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	GRDECINBF1						
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 29.57 GRDECINBF1 Register Contents

Bit Position	Bit Name	Function
31 to 7	—	Reserved These bits are always read as 0. The write value should also be 0.
6 to 0	GRDECINBF1	These bits are valid when ECCTST = 1 (selecting test mode) in the global RAM test control register. When DECINEN = 1, the value of this register is input to the ECC decoder as data for use in updating in response to the execution of an RMW instruction. The value is treated as if it were 7 bits of data from ECC. This register is common to banks A and B and provides the values for both the 32 higher-order bits and the 32 lower-order bits.

(7) GRERRINT_VCI/_PE1 — Global RAM Error Information Control Register

GRERRINT enables or disables generation of the error notification signal to the ECM upon detection of a 2-bit ECC error, a 1-bit ECC error, or an address parity error.

The setting of this register is used for accesses through the respective access port.

GRERRINT is initialized by an internal reset or an external reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	APEIE	DEDIE	SEDIE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W

Table 29.58 GRERRINT_VCI/PE1 Register Contents

Bit Position	Bit Name	Function
31 to 3	—	Reserved These bits are always read as 0. The write value should also be 0.
2	APEIE	Address Parity Error Notification Enable Enables or disables generation of the error notification signal upon detection of an address parity error when address parity check is enabled. 0: Disables notification of the address parity error. 1: Enables notification of the address parity error.
1	DEDIE	2-Bit ECC Error Notification Enable Enables or disables generation of the error notification signal upon detection of a 2-bit error when ECC error detection and correction are enabled. 0: Disables notification of the 2-bit ECC error. 1: Enables notification of the 2-bit ECC error.
0	SEDIE	1-Bit ECC Error Notification Enable Enables or disables generation of the error notification signal upon detection of a 1-bit error when ECC error detection and correction are enabled. 0: Disables notification of the 1-bit ECC error. 1: Enables notification of the 1-bit ECC error.

(8) GRSTCLR_VCI/_PE1 — Global RAM Status Clear Register

GRSTCLR clears the error flags in the error status register (GR1STERSTR), the overflow flag in the error count overflow status register (GROVFSTR), and the error address register (GR1STEADR). GRSTCLR is a write-only register and is always read as 0.

The setting of this register is used for accesses through the respective access port.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	STCLR1	STCLR0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	W	W

Table 29.59 GRSTCLR_VCI/PE1 Register Contents

Bit Position	Bit Name	Function
31 to 2	—	Reserved These bits are always read as 0. The write value should also be 0.
1	STCLR1	Error Overflow Flag Clear (for the upper 32-bit data) Writing 1 to this bit clears the EXDEDF1, EXSEDF1, APEF1, DEDF1, and SEDF1 flags in GR1STERSTR; ERROVF1 flag in GROVFSTR; and GR1STEADR1.
0	STCLR0	Error Overflow Flag Clear (for the lower 32-bit data) Writing 1 to this bit clears the EXDEDF0, EXSEDF0, APEF0, DEDF0, and SEDF0 flags in GR1STERSTR; ERROVF0 flag in GROVFSTR; and GR1STEADR0.

(9) GROVFSTR_VCI/PE1 — Global RAM Error Count Overflow Status Register

GROVFSTR monitors occurrence of error overflow. If the second error occurs after the first error (= while any of the error flags in the error status register is set), the flag in this register is set. However, if the second error is identical to the first error (both the source and address are same), this flag is not set. ERROVF is cleared by an internal reset, an external reset, or setting the STCLR bit to 1 in GRSTCLR.

The setting of this register is used for accesses through the respective access port.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ERROVF1	ERROVF0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 29.60 GROVFSTR_VCI/PE1 Register Contents

Bit Position	Bit Name	Function
31 to 2	—	Reserved These bits are always read as 0. The write value should also be 0.
1	ERROVF1	Error Overflow Flag (for the upper 32-bit data) This flag is set if the second error occurs while any of the error flags (EXDEDF1, EXSEDF1, APEF1, DEDF1, and SEDF1) in the error status register is set, except when both of the error address and source of the second error are the same as those of the first error.
0	ERROVF0	Error Overflow Flag (for the lower 32-bit data) This flag is set if the second error occurs while any of the error flags (EXDEDF0, EXSEDF0, APEF0, DEDF0, and SEDF0) in the error status register is set, except when both of the error address and source of the second error are the same as those of the first error.

(10) GR1STERSTR_VCI/PE1 — Global RAM 1st Error Status Register

GR1STERSTR monitors occurrence of the first error. Errors are detected and the states are updated independently for the higher- and lower-order 32 bits.
An error status flag is set if an error occurs while all error flags for the higher-order or the lower-order bits are 0. Note, however, that the corresponding error flag is also set if a 2-bit ECC error or an address parity error occurs in subsequent access as long as only the 1-bit ECC error flag is set at the time.

If more than one error is detected for a given memory access, all the corresponding error flags are set. GR1STERSTR is cleared by an internal reset, an external reset, or setting the STCLR bit to 1 in GRSTCLR.

The setting of this register is used for accesses through the respective access port.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	EXDEDF1	EXSEDF1	—	APEF1	DEDF1	SEDF1	—	—	EXDEDF0	EXSEDF0	—	APEF0	DEDF0	SEDF0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 29.61 GR1STERSTR_VCI/PE1 Register Contents (1/2)

Bit Position	Bit Name	Function
7+8n 6+8n	—	Reserved These bits are always read as 0. The write value should also be 0.
5+8n	EXDEDFn	2-Bit ECC Error Monitor Flag Clearing condition: Cleared to 0 by a reset or by setting the STCLRn bit to 1 in GRSTCLR. Setting condition: A 2-bit ECC error has occurred while the error flags EXDEDFn, APEFn, and DEDFn are 0 (during RMW processing for Global RAM).
4+8n	EXSEDFn	1-Bit ECC Error Monitor Flag Clearing condition: Cleared to 0 by a reset or by setting the STCLRn bit to 1 in GRSTCLR. Setting condition: A 1-bit ECC error has occurred while the error flags EXDEDFn, EXSEDFn, APEFn, DEDFn, and SEDFn are 0 (during RMW processing for Global RAM).
3+8n	—	Reserved These bits are always read as 0. The write value should also be 0.
2+8n	APEFn	Address Parity Error Monitor Flag Clearing condition: Cleared to 0 by a reset or by setting the STCLRn bit to 1 in GRSTCLR. Setting condition: An address parity error has occurred while the error flags EXDEDFn, APEFn, and DEDFn are 0. **NOTE** This flag is set equally for read and write; the error generation source is not considered.

Table 29.61 GR1STERSTR_VCI/PE1 Register Contents (2/2)

Bit Position	Bit Name	Function
1+8n	DEDFn	2-Bit ECC Error Monitor Flag Clearing condition: Cleared to 0 by a reset or by setting the STCLRn bit to 1 in GRSTCLR. Setting condition: A 2-bit ECC error has occurred while the error flags EXDEDFn, APEFn, and DEDFn are 0.
0+8n	SEDFn	1-Bit ECC Error Monitor Flag Clearing condition: Cleared to 0 by a reset or by setting the STCLRn bit to 1 in GRSTCLR. Setting condition: A 1-bit ECC error has occurred while the error flags EXDEDFn, EXSEDFn, APEFn, DEDFn, and SEDFn are 0.

Note 1. n = 0, 1, n = 0 represents the 32 lower-order bits and n = 1 represents the 32 higher-order bits.

(11) GR1STEADRn_VCI/PE1 — Global RAM 1st Error Address Register n (n = 0, 1)

GR1STEADRn holds the address at which an error has occurred.

The error address is set if an error occurs while all the error flags are 0 in GR1STERSTR. The address is updated if a 2-bit ECC error or an address parity error occurs while the 1-bit ECC error flag is set as the first error. Once a 2-bit ECC error or an address parity error occurs, the address is not updated.

EADR[20:0] of this register correspond to bits [20:0] of the real address. The real address can be calculated by appending the higher-order address bits [31:21] as a base address.

GR1STEADR is cleared by an internal reset, an external reset, or setting the STCLR bit to 1 in GRSTCLR. During accesses to the lower 32-bit data, the address is stored in GR1STEADR0. During accesses to the 32 higher-order bits data, the address is stored in GR1STEADR1.

The setting of this register is used for accesses through the respective access port.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	EADR[20:16]				
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	EADR[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 29.62 GR1STEADRn_VCI/PE1 Register Contents

Bit Position	Bit Name	Function
31 to 21	—	Reserved These bits are always read as 0. The write value should also be 0.
20 to 0	EADR[20:0]	1st Error Address These bits monitor the address of the first error. The error address is set if an error occurs while all the error flags are 0 in GR1STERSTR. The address is updated if a 2-bit ECC error or an address parity error occurs while the 1-bit ECC error flag is set as the first error. Once a 2-bit ECC error or an address parity error occurs, the address is not updated.

29.2.6.4 Test Function

Through appropriate register setting, desired values can be written as RAM data and to the ECC and address parity bits. Also, data in the RAM and the ECC and address parity bits and the ECC decoder output data for RMW can all be read.

It is possible to input the desired data in the ECC decoder for RMW.

(1) Writing RAM Data

(a) Set the ECCTST bit in the global RAM test control register to 1 to set test mode.

(b) Set the corresponding DATSEL1 to 0 and DATSEL0 to 0 in the global RAM test control register to select RAM data for access when writing.

(c) When data is written to the global RAM, only the RAM data can be modified without updating the ECC bits.

How to exit this test mode:

(a) Set the ECCTST bit in the global RAM test control register to 0 to disable test mode (normal mode).

(2) Reading RAM Data

(a) Set the ECCDIS bit in the global RAM ECC control register to 1 to disable ECC error detection and correction.

(b) Read the global RAM. Since neither error detection nor correction proceeds when the global RAM is read, the RAM data is read unchanged.

How to exit this test mode:

(a) Set the ECCDIS bit in the global RAM ECC control register to 0 to enable ECC error detection and correction.

(3) Writing to the ECC and Address Parity Bits

(a) Set the ECCTST bit in the global RAM test control register to 1 to set test mode.

(b) Set the corresponding DATSEL1 bit to 0 and DATSEL0 bit to 1 in the global RAM test control register to select the ECC bits and the address parity bit for access when writing.

(c) When data is written to the global RAM, only the ECC and address parity bits can be modified without updating the RAM data. At that time, bit[6:0] and bit[7] are respectively written to the ECC bits and to the address parity bit.

NOTE

Writing desired values to the RAM data and the address parity bit should be in order of 1) writing the RAM data and then 2) writing the address parity bit.
When writing desired values to the RAM data and ECC bits, you can start either by writing to the RAM data or ECC bits.

How to exit this test mode:

(a) Set the ECCTST bit in the local RAM test control register to 0 to disable test mode (normal mode).

(4) Reading the ECC and Address Parity Bits

(a) Set the ECCTST bit in the global RAM test control register to 1 to set test mode.

(b) Setting the DATSEL1 and DATSEL0 bits in the global RAM test control register to 0 and 1, respectively, to select the ECC and address parity bits for access when reading.

(c) When data in the global RAM is read, the ECC and address parity bits are stored in the corresponding register from among global RAM test data read buffers 0 to 3.

How to exit this test mode:

(a) Set the ECCTST bit in the global RAM test control register to 0 to disable test mode (normal mode).

(5) Self-Diagnosis of the ECC Check Function for the Access Ports

Desired values can be written to the RAM data and/or ECC bits by following the procedure described in (1) or (3) above. Therefore, a fault can be injected by, for example, inverting appropriate bits of the RAM data and/or ECC bits. After that, self-diagnosis of the ECC decoder is possible by reading the global RAM in normal mode and checking the result of error correction or detection.

(6) Self-Diagnosis of the Address Parity Check Function

(a) Self-diagnosis is enabled by following either of the two procedures below.

(b) Setting APTESTi (i = 0, 1, 2, 3) in the global RAM test control register to 1 inverts the result of address parity generation for the corresponding data area (the 32 higher-order bits or the 32 lower-order bits) of the corresponding bank (bank A or B). That is, a fault can be injected to the address parity generator. Writing to the corresponding data area of the corresponding bank in the global RAM in this state and checking the result of parity error detection allows self-diagnosis of the address parity check function in writing.

(c) Desired data can be written to the address parity bits through the procedure described in (4) above. This enables injection of a fault to the address parity bit by inverting the address parity bit. After that, self-diagnosis of the address parity bit checking function for reading is possible by reading the global RAM in normal mode and checking the result of parity error detection.

(7) Self-Diagnosis of the ECC Decoder for the Data Read Out in an RMW Operation

(a) Suitable erroneous values are injected as RAM data or to the ECC bits by following procedure (1) or (3) above.

(b) Setting the DATSEL1 and DATSEL0 bits in the test control registers for the global RAM to 1 and 0, respectively, makes the output data from the ECC decoder for the data read out in an RMW operation the target for reading.

(c) After RMW processing for the global RAM proceeds, the data read out in an RMW operation for global RAM are stored in the corresponding register from among global RAM test data read buffers 0 to 3. Checking the result allows self-diagnosis of the ECC decoder for the data read out in an RMW operation.

(8) Self-Diagnosis of the ECC Decoder for the Data being Updated at the Time of RMW Access

(a) Setting the DATSEL1 and DATSEL0 bits in the global RAM test control register to 1 makes the output data from the ECC decoder for the data being updated at the time of RMW access the target for reading.

(b) Through the above setting, the input data from the ECC decoder for the data being updated at the time of RMW access is switched to ECC decoder input buffer 0 or 1 (GRDECINBF0 or 1) from write data sent from the access ports. As a result, suitable erroneous values can be injected by setting an appropriate value in ECC decoder input buffer 0 or 1.

(c) After RMW processing for the global RAM proceeds, the data being updated at the time of RMW access for global RAM are stored in the corresponding register from among global RAM test data read buffers 0 to 3. Checking the result allows self-diagnosis of the ECC decoder when data are updated.

29.2.7 Instruction Cache ECC and EDC

29.2.7.1 Overview

The instruction cache ECC is summarized in the table below.

Table 29.63 Overview of Instruction Cache ECC

Item	Description
ECC error detection and correction	ECC error detection and correction can be either enabled or disabled. When enabled, either of the following settings can be selected. • ECC error detection and correction are carried out (2-bit error detection and 1-bit error detection and correction are carried out). • ECC error detection is carried out (2-bit error detection and 1-bit error detection are carried out). When disabled, neither error detection nor correction is carried out. In the state after reset, 1-bit error detection and correction and 2-bit error detection are carried out while the ECC function is enabled.
Address parity	None
Error notification	Upon occurrence of an ECC error, the ECM is notified of the error. **ECC Error:** • Error notification can be either enabled or disabled upon detection of a 2-bit ECC error. • Error notification can be either enabled or disabled upon detection of a 1-bit ECC error. In the state after reset, notification of the 2-bit ECC error is disabled, and notification of the 1-bit ECC error is disabled. The error notification signal is output, where a 2-bit ECC error is handled as one source, and a 1-bit ECC error is handled as one source.
Error status	A status register is provided, which indicates the statuses of 2-bit ECC error detection and 1-bit ECC error detection. If an error occurs while no error status flag is set, the corresponding status is set. The error status flag can be cleared using the clear register.
Address capture	If an ECC error occurs while no error status flag is set, the address at which the associated error has occurred is captured. The address is captured when a 2-bit ECC error or a 1-bit ECC error is detected. The error status flag serves as the enable bit of the capture address.
Self-diagnosis	A cache instruction is used to write the desired values as RAM data and to the ECC bits. Similarly, data in the RAM and the ECC bits can be read directly. Since instructions as described above go through the same encoding or decoding path as a normal cache fill or instruction fetch, only such instructions can be used in inserting and confirming errors.

29.2.7.2 List of Registers

Table 29.64 List of Registers

Address	Symbol	Register Name	R/W	Value after Reset	Access Size
FFC6 0400$_H$	IDECCCTL_PE1	Instruction cache data RAM ECC control register (PE1)	R/W	0000 0000$_H$	16/32
FFC6 0404$_H$	IDERRINT_PE1	Instruction cache data RAM error information control register (PE1)	R/W	0000 0000$_H$	8/16/32
FFC6 0408$_H$	IDSTCLR_PE1	Instruction cache data RAM error status clear register (PE1)	W	0000 0000$_H$	8/16/32
FFC6 040C$_H$	IDOVFSTR_PE1	Instruction cache data RAM error count overflow status register (PE1)	R	0000 0000$_H$	8/16/32
FFC6 0410$_H$	ID1STERSTR_PE1	Instruction cache data RAM 1st error status register (PE1)	R	0000 0000$_H$	8/16/32
FFC6 0450$_H$	ID1STEADR0_PE1	Instruction cache data RAM (bank0) 1st error address register (PE1)	R	0000 0000$_H$	8/16/32
FFC6 0454$_H$	ID1STEADR1_PE1	Instruction cache data RAM (bank1) 1st error address register (PE1)	R	0000 0000$_H$	8/16/32
FFC6 1400$_H$	ITECCCTL_PE1	Instruction cache tag RAM ECC control register (PE1)	R/W	0000 0000$_H$	16/32
FFC6 1404$_H$	ITERRINT_PE1	Instruction cache tag RAM error information control register (PE1)	R/W	0000 0000$_H$	8/16/32
FFC6 1408$_H$	ITSTCLR_PE1	Instruction cache tag RAM error status clear register (PE1)	W	0000 0000$_H$	8/16/32
FFC6 140C$_H$	ITOVFSTR_PE1	Instruction cache tag RAM error count overflow status register (PE1)	R	0000 0000$_H$	8/16/32
FFC6 1410$_H$	IT1STERSTR_PE1	Instruction cache tag RAM 1st error status clear register (PE1)	R	0000 0000$_H$	8/16/32
FFC6 1450$_H$	IT1STEADR0_PE1	Instruction cache tag RAM (bank0) 1st error address register (PE1)	R	0000 0000$_H$	8/16/32

29.2.7.3 Details of Registers

(1) IDECCCTL_PE1 — Instruction Cache Data RAM ECC Control Register

IDECCCTL enables or disables ECC error detection and correction and 1-bit error correction for cache data RAM. Set PROT[1:0] bits to 01_B when writing to this register.

This register is initialized by an internal reset or an external reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PROT[1:0]		—	—	—	—	—	—	—	—	—	—	—	—	SECDIS	ECCDIS
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R\W	R\W

Table 29.65 IDECCCTL_PE1 Register Contents

Bit Position	Bit Name	Function
31 to 16	—	Reserved These bits are always read as 0. The write value should also be 0.
15, 14	PROT[1:0]	These bits enable to write to this register. The read value is 0.
13 to 2	—	Reserved These bits are always read as 0. The write value should also be 0.
1	SECDIS	1-Bit Error Correction Disable Enables or disables 1-bit error correction when ECC error detection and correction are enabled. 0: Enables correction of the 1-bit error detected. 1: Disables correction of the 1-bit error detected.
0	ECCDIS	ECC Disable Enables or disables ECC error detection and correction. 0: Enables ECC error detection and correction. 1: Disables ECC error detection and correction.

(2) IDERRINT_PE1 — Instruction Cache Data RAM Error Information Control Register

IDERRINT enables or disables generation of the error notification signal to the ECM upon detection of a 2-bit ECC error or a 1-bit ECC error in cache data RAM.

IDERRINT is initialized by an internal reset or an external reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DEDIE	SEDIE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Table 29.66 IDERRINT_PE1 Register Contents

Bit Position	Bit Name	Function
31 to 2	—	Reserved These bits are always read as 0. The write value should also be 0.
1	DEDIE	2-Bit ECC Error Notification Enable Enables or disables generation of the error notification signal upon detection of a 2-bit error when ECC error detection and correction are enabled. 0: Disables notification of the 2-bit ECC error. 1: Enables notification of the 2-bit ECC error.
0	SEDIE	1-Bit ECC Error Notification Enable Enables or disables generation of the error notification signal upon detection of a 1-bit error when ECC error detection and correction are enabled. 0: Disables notification of the 1-bit ECC error. 1: Enables notification of the 1-bit ECC error.

(3) IDSTCLR_PE1 — Instruction Cache Data RAM Error Status Clear Register

IDSTCLR clears the error flags in the error status register (ID1STERSTR), the overflow flag in the error count overflow status register (IDOVFSTR), and the error address register (ID1STEADR). IDSTCLR is a write-only register and is always read as 0.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	STCLR1	STCLR0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	W	W

Table 29.67 IDSTCLR_PE1 Register Contents

Bit Position	Bit Name	Function
31 to 2	—	Reserved These bits are always read as 0. The write value should also be 0.
1	STCLR1	Error Overflow Flag Clear (for bank 1) Writing 1 to this bit clears the DEDF1and SEDF1 flags in ID1STERSTR; ERROVF1 flag in IDOVFSTR; and ID1STEADR1.
0	STCLR0	Error Overflow Flag Clear (for bank 0) Writing 1 to this bit clears the DEDF0and SEDF0 flags in ID1STERSTR; ERROVF0 flag in IDOVFSTR; and ID1STEADR0.

(4) IDOVFSTR_PE1 — Instruction Cache Data RAM Error Count Overflow Status Register

IDOVFSTR monitors occurrence of error overflow in cache data RAM. If the second error occurs after the first error (= while any of the error flags in the error status register is set), the flag in this register is set. However, if the second error is identical to the first error (both the source and address are same), this flag is not set. ERROVF is cleared by an internal reset, an external reset, or setting the STCLR bit to 1 in IDSTCLR.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ERROVF1	ERROVF0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 29.68 IDOVFSTR_PE1 Register Contents

Bit Position	Bit Name	Function
31 to 2	—	Reserved These bits are always read as 0. The write value should also be 0.
1	ERROVF1	Error Overflow Flag (for bank 1) This flag is set if the second error occurs while any of the error flags (DEDF1 and SEDF1) in the error status register is set, except when both of the error address and source of the second error are the same as those of the first error.
0	ERROVF0	Error Overflow Flag (for bank 0) This flag is set if the second error occurs while any of the error flags (DEDF0 and SEDF0) in the error status register is set, except when both of the error address and source of the second error are the same as those of the first error.

(5) ID1STERSTR_PE1 — Instruction Cache Data RAM 1st Error Status Register

ID1STERSTR monitors occurrence of the first error in cache data RAM. The error status flag is set if an error occurs while all the error flags in the same bank are 0. The corresponding error flag is set if a 2-bit ECC error occurs while the 1-bit ECC error flag is set.

This register cannot be used to identify the way that had an error.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	DEDF1	SEDF1	—	—	—	—	—	—	DEDF0	SEDF0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 29.69 ID1STERSTR_PE1 Register Contents

Bit Position	Bit Name	Function
31 to 10	—	Reserved These bits are always read as 0. The write value should also be 0.
9	DEDF1	2-Bit ECC Error Monitor Flag (for bank 1) Clearing condition: Cleared to 0 by a reset or by setting the STCLR1 bit to 1 in IDSTCLR. Setting condition: A 2-bit ECC error has occurred while the error flag DEDF1 is 0.
8	SEDF1	1-Bit ECC Error Monitor Flag (for bank 1) Clearing condition: Cleared to 0 by a reset or by setting the STCLR1 bit to 1 in IDSTCLR. Setting condition: A 1-bit ECC error has occurred while the error flags DEDF1 and SEDF1 are 0.
7 to 2	—	Reserved These bits are always read as 0. The write value should also be 0.
1	DEDF0	2-Bit ECC Error Monitor Flag (for bank 0) Clearing condition: Cleared to 0 by a reset or by setting the STCLR0 bit to 1 in IDSTCLR. Setting condition: A 2-bit ECC error has occurred while the error flag DEDF0 is 0.
0	SEDF0	1-Bit ECC Error Monitor Flag (for bank 0) Clearing condition: Cleared to 0 by a reset or by setting the STCLR0 bit to 1 in IDSTCLR. Setting condition: A 1-bit ECC error has occurred while the error flags DEDF0 and SEDF0 are 0.

(6) ID1STEADRn_PE1 — Instruction Cache Data RAM (Bank n) 1st Error Address Register (n = 0, 1)

ID1STEADR holds the address at which an error has occurred in cache data RAM.

The error address is set if an error occurs while all the error flags for the relevant banks are 0 in ID1STERSTR. The address is updated if a 2-bit ECC error is found while the 1-bit ECC error flag is set for a first error. Once a 2-bit ECC error occurs, the address is not updated.

EADRn[6:0] indicate the cache address [10:4]. EADRn[7] is always 0. EADRn[8] indicates the way group number.

ID1STEADR is cleared by an internal reset, an external reset, or setting the STCLR bit to 1 in IDSTCLR.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	EADRn[8:0]								
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 29.70 ID1STEADRn_PE1 Register Contents

Bit Position	Bit Name	Function
31 to 9	—	Reserved These bits are always read as 0. The write value should also be 0.
8 to 0	EADRn[8:0]	1st Error Address (for bank n) These bits monitor the address of the first error. The error address is set if an error occurs while all the error flags for bank n are 0 in ID1STERSTR. The address is updated if a 2-bit ECC error is found while the 1-bit ECC error flag is set as the first error. Once a 2-bit ECC error occurs, the address is not updated.

(7) ITECCCTL_PE1 — Instruction Cache Tag RAM ECC Control Register

This register is used to enable or disable ECC error detection of the cache TAG RAM. Set PROT[1:0] bits to 01_B when writing to this register.

This register is initialized by an internal reset or an external reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PROT[1:0]		—	—	—	—	—	—	—	—	—	—	—	—	—	ECCDIS
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 29.71 ITECCCTL_PE1 Register Contents

Bit Position	Bit Name	Function
31 to 16	—	Reserved These bits are always read as 0. The write value should also be 0.
15, 14	PROT[1:0]	These bits enable to write to this register. The read value is 0.
13 to 1	—	Reserved These bits are always read as 0. The write value should also be 0.
0	ECCDIS	ECC Disable Enables or disables ECC error detection. 0: Enables ECC error detection. 1: Disables ECC error detection.

(8) ITERRINT_PE1 — Instruction Cache Tag RAM Error Information Control Register

ITERRINT enables or disables generation of the error notification signal to the ECM upon detection of a 2-bit ECC error or a 1-bit ECC error in cache tag RAM.

ITERRINT is initialized by an internal reset or an external reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DEDIE	SEDIE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Table 29.72 ITERRINT_PE1 Register Contents

Bit Position	Bit Name	Function
31 to 2	—	Reserved These bits are always read as 0. The write value should also be 0.
1	DEDIE	2-Bit ECC Error Notification Enable Enables or disables generation of the error notification signal upon detection of a 2-bit error when ECC error detection and correction are enabled. 0: Disables notification of the 2-bit ECC error. 1: Enables notification of the 2-bit ECC error.
0	SEDIE	1-Bit ECC Error Notification Enable Enables or disables generation of the error notification signal upon detection of a 1-bit error when ECC error detection and correction are enabled. 0: Disables notification of the 1-bit ECC error. 1: Enables notification of the 1-bit ECC error.

(9) ITSTCLR_PE1 — Instruction Cache Tag RAM Error Status Clear Register

ITSTCLR clears the error flags in the error status register (IT1STERSTR), the overflow flag in the error count overflow status register (ITOVFSTR), and the error address register (IT1STEADR0). ITSTCLR is a write-only register and is always read as 0.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	STCLR0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	W

Table 29.73 ITSTCLR_PE1 Register Contents

Bit Position	Bit Name	Function
31 to 1	—	Reserved These bits are always read as 0. The write value should also be 0.
0	STCLR0	Error Overflow Flag Clear (for bank 0) Writing 1 to this bit clears the DEDF0 and SEDF0 flags in IT1STERSTR; ERROVF0 flag in ITOVFSTR; and IT1STEADR0.

(10) ITOVFSTR_PE1 — Instruction Cache Tag RAM Error Count Overflow Status Register

ITOVFSTR monitors occurrence of error overflow in cache tag RAM. If the second error occurs after the first error (= while any of the error flags in the error status register is set), the flag in this register is set. However, if the second error is identical to the first error (both the source and address are same), this flag is not set. ERROVF is cleared by an internal reset, an external reset, or setting the STCLR bit to 1 in ITSTCLR.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ERROVF0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 29.74 ITOVFSTR_PE1 Register Contents

Bit Position	Bit Name	Function
31 to 1	—	Reserved These bits are always read as 0. The write value should also be 0.
0	ERROVF0	Error Overflow Flag (for bank 0) This flag is set if the second error occurs while any of the error flags (DEDF0 and SEDF0) in the error status register is set, except when both of the error address and source of the second error are the same as those of the first error.

(11) IT1STERSTR_PE1 — Instruction Cache Tag RAM 1st Error Status Register

IT1STERSTR monitors occurrence of the first error in cache tag RAM.

The error status flag is set if an error occurs while all the error flags are 0. The corresponding error flag is set if a 2-bit ECC error occurs while the 1-bit ECC error flag is set.

This register cannot be used to identify the way that had an error.

IT1STERSTR is cleared by an internal reset, an external reset, or setting the STCLR bit to 1 in ITSTCLR.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	DEDF0	SEDF0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 29.75 IT1STERSTR_PE1 Register Contents

Bit Position	Bit Name	Function
31 to 2	—	Reserved These bits are always read as 0. The write value should also be 0.
1	DEDF0	2-Bit ECC Error Monitor Flag Clearing condition: Cleared to 0 by a reset or by setting the STCLR0 bit to 1 in ITSTCLR. Setting condition: A 2-bit ECC error has occurred while the error flag DEDF0 is 0.
0	SEDF0	1-Bit ECC Error Monitor Flag Clearing condition: Cleared to 0 by a reset or by setting the STCLR0 bit to 1 in ITSTCLR. Setting condition: A 1-bit ECC error has occurred while the error flags DEDF0 and SEDF0 are 0.

(12) IT1STEADR0_PE1 — Instruction Cache Tag RAM 1st Error Address Register

IT1STEADR0 holds the address at which an error has occurred in cache tag RAM.

The error address is set if an error occurs while all the error flags are 0 in IT1STERSTR. The address is updated if a 2-bit ECC error is found while the 1-bit ECC error flag is set as the first error. Once a 2-bit ECC error occurs, the address is not updated.

EADR[6:0] indicate the cache address [10:4]. EADR[7] is always 0. EADRn[8] indicates the way group number.

IT1STEADR0 is cleared by an internal reset, an external reset, or setting the STCLR bit to 1 in ITSTCLR.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	EADR[8:0]								
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 29.76 IT1STEADR0_PE1 Register Contents

Bit Position	Bit Name	Function
31 to 9	—	Reserved These bits are always read as 0. The write value should also be 0.
8 to 0	EADR[8:0]	1st Error Address Monitors the address of the first error. The error address is set if an error occurs while all the error flags are 0 in IT1STERSTR. The address is updated if a 2-bit ECC error is found while the 1-bit ECC error flag is set as the first error. Once a 2-bit ECC error or an address parity error occurs, the address is not updated.

29.2.7.4 Test Function

A cache instruction is used to write the desired values as RAM data and to the ECC bits, and read data in the RAM and the ECC bits directly.

Since instructions as described above go through the same encoding or decoding path as a normal cache fill or instruction fetch, only such instructions can be used in inserting and confirming errors.

For details, see RH850G3M User's Manual: Software.

29.2.8 DTS RAM ECC

See **Section 7, DMA**.

29.2.9 ECC for Peripheral RAM (32 Bits)

29.2.9.1 Overview

This is an ECC module for the RAM of the following peripheral modules.

RS-CAN, FlexRay, APA, and DFE

Error Detection and Correction

Seven-bit ECC data is appended to the 32-bit RAM data.

This ECC module provides 2-bit ECC error detection and 1-bit ECC error detection and correction.

CAUTION

This module is not capable of reliably detecting errors in three or more bits.

If errors occur in three or more bits, the module may detect the errors as 1- or 2-bit ECC errors or not detect any errors. Depending on the settings, this may lead to the correction of a bit that was not actually inverted.

Enabling or Disabling ECC Error Detection and Correction

- ECC error detection can be either enabled or disabled.
- One-bit ECC error correction can be either enabled or disabled.
- If all the bits of RAM output data are stuck to 0 or 1, it is detected as a 2-bit ECC error.

Error Notification

- The ECM is notified when 2-bit ECC errors are detected (this can be enabled or disabled).
- The ECM is notified when 1-bit ECC errors are detected (this can be enabled or disabled).

Once the ECM has been notified of an error, even if another ECC error is detected, the ECM is not notified until the error status bit corresponding to the initial error is cleared.

Error Status

- Detection of 2-bit and 1-bit ECC errors can be monitored.
- Special registers are provided to clear error status.

Address Capture

- Only one address at which an ECC error has occurred can be captured.
- When a 2-bit or 1-bit ECC error is detected, the error-causing address is captured. It is used to capture only when the first error is detected after the flag is cleared.

Testing Function (Error Insertion)

- By setting the test mode, register values can be used as the data to be output to the RAM. When a peripheral module writes to the RAM, the ECEDB[31:0] register value can be written to the RAM data section, and the ECERDB[6:0] register value can be written to the ECC redundant bit section.
- By setting the test mode, the ECC redundant bit section can be latched when RAM data is read, and the value can be confirmed.
- By setting the test mode, the ECC redundant bit (encoding circuit) and syndrome code (decoding circuit), which are generated from the input data, can be confirmed.

29.2.9.2 List of Registers

(1) List of ECC Modules

The RAMs of the multiple peripheral functions are provided with the ECC modules. The following table shows the peripheral functions provided with the ECC modules, the corresponding ECC module names, and base addresses of the ECC modules.

Table 29.77 Peripheral Functions, Corresponding ECC Module Names, and Base Addresses of ECC Modules

Peripheral Functions		ECC Module Names and Register Base Addresses			
		Master Side*1		Checker Side*1	
		Module Name	Base Address <base_addr>	Module Names	Base Address <base_addr>
RS-CAN		E7RC0M	FFC7 1000_H	E7RC0C	FFC7 1200_H
FlexRay	Message RAM (MRAM)	E7FR0M	FFC7 2000_H	E7FR0C	FFC7 2200_H
	Temporary buffer (TBF AA)	E7FR1M	FFC7 2400_H	E7FR1C	FFC7 2600_H
	Temporary buffer (TBF B)	E7FR2M	FFC7 2800_H	E7FR2C	FFC7 $2A00_H$
APA	Element RAM0 (ELRAM0)	E7AP0M	FFC7 6000_H	E7AP0C	FFC7 6200_H
	Element RAM1 (ELRAM1)	E7AP1M	FFC7 6400_H	E7AP1C	FFC7 6600_H
	Element RAM2 (ELRAM2)	E7AP2M	FFC7 6800_H	E7AP2C	FFC7 $6A00_H$
DFE	Coefficient memory (CMEM)	E7DF1M	FFC7 $D000_H$	E7DF1C	FFC7 $D200_H$

Note 1. Two ECC modules are provided to support BIST, one for the master and the other for the checker. For details, refer to **Section 29.7, BIST**.

(2) List of Registers

Each ECC module has the registers shown in the following table.

Table 29.78 List of ECC Module Registers

Register Name	Symbol	R/W	Value after Reset	Address	Access Size
ECC control register*1	E710CTL	R/W	$001X_H$	<base_addr> + 00_H	16/8
ECC test mode control register	E710TMC	R/W	0000_H	<base_addr> + 04_H	16/8
ECC redundant bit data control test register	E710TRC	R/W	0000 0000_H	<base_addr> + 08_H	32
ECC encoder and decoder data test register	E710TED	R/W	0000 0000_H	<base_addr> + $0C_H$	32
ECC error address register	E710EAD	R/W	0000 0000_H	<base_addr> + 10_H	32

Note 1. The reset value of the LSB in the ECC control register is undefined.

(3) Register Map

Table 29.79 List of Register Map

Symbol	31 24	23 16	15 8	7 0	Address
E710CTL	— (00_H)	— (00_H)	ECCTL[15:8]	ECCTL[7:0]	$nn00_H$
E710TMC	— (00_H)	— (00_H)	ECTMC[15:8]	ECTMC[7:0]	$nn04_H$
E710TRC	ECSYND[7:0]	ECHORD[7:0]	ECECRD[7:0]	ECERDB[7:0]	$nn08_H$
E710TED	ECEDB[31:24]	ECEDB[23:16]	ECEDB[15:8]	ECEDB[7:0]	$nn0C_H$
E710EAD	ECEAD[31:24]	ECEAD[23:16]	ECEAD[15:8]	ECEAD[7:0]	$nn10_H$

29.2.9.3 Details of Registers

(1) E710CTL — ECC Control Register

E710CTL controls the status and modes of the ECC module.

E710CTL can be read and written to using the 16-bit or 8-bit manipulation instruction.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	EMCA[1:0]		—	—	—	ECER2C	ECER1C	—	ECTHM	—	EC1ECP	EC2EDIC	EC1EDIC	ECER2F	ECER1F	ECEMF
Value after reset	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	Un-defined
R/W	R/W*1	R/W*1	R	R	R	R/W*1	R/W*1	R	R/W	R	R/W	R/W	R/W	R	R	R

Note 1. This bit is always read as 0.

Table 29.80 E710CTL Register Contents (1/2)

Bit Position	Bit Name	Function
15, 14	EMCA[1:0]	Access Control 1 and 0 to ECC Mode Select Bit These bits reserve the write trigger to bit 7. The read value is always 0.
10	ECER2C	2-Bit ECC Error Detection Flag Clear Clears the bit 2 (ECER2F) status flag. The read value is always 0, and writing 0 to ECER2C does not change the state. If writing 1 to ECER2C conflicts with the condition for setting bit 2, the former takes priority. Writing 1 to ECER2C while ECER2F is set clears ECER2F.
9	ECER1C	1-Bit ECC Error Detection Flag Clear Clears the bit 1 (ECER1F) status flag. The read value is always 0, and writing 0 to ECER1C does not change the state. If writing 1 to ECER1C conflicts with the condition for setting bit 1, the former takes priority. Writing 1 to ECER1C while ECER1F is set clears ECER1F.
7	ECTHM	ECC Function Disable Select Selects the ECC decoding operation. Write access to ECTHM is enabled when the value of bits 15 and 14 is 01_B. Therefore, only the 16-bit manipulation instruction is valid. Setting ECTHM to 1 disables error detection and bit correction. Here, if the data to be output to the peripheral module contains an error, the data is output without bit correction. Setting ECTHM to 1 has no effect on the encoder side. 0: Enables ECC detection and correction. 1: Disables ECC detection and correction.
5	EC1ECP	1-Bit ECC Error Correction Enable 0: Enables 1-bit error correction upon error detection. 1: Disables 1-bit error correction upon error detection.
4	EC2EDIC	2-Bit ECC Error Detection Notification Enable 0: When a 2-bit error is detected, the ECM is not notified of the error. 1: When a 2-bit error is detected, the ECM is notified of the error.
3	EC1EDIC	1-Bit ECC Error Detection Notification Enable 0: When a 1-bit error is detected, the ECM is not notified of the error. 1: When a 1-bit error is detected, the ECM is notified of the error.
2	ECER2F	2-Bit ECC Error Detection Flag Indicates that errors have been detected at two bits in bits 0 to 38 of the data read from the RAM during RAM read access while error detection is enabled. This bit is a read only flag. 0: 2-bit error is not generated 1: 2-bit error is generated Clearing conditions (1) Reset is applied. (2) 1 is written to ECER2C. (3) ECC detection and correction being disabled (ECTHM = 1).

Table 29.80 E710CTL Register Contents (2/2)

Bit Position	Bit Name	Function
1	ECER1F	1-Bit ECC Error Detection And Correction Flag Indicates that an error has been detected at one bit in bits 0 to 38 of the data read from the RAM during RAM read access while error detection is enabled. This bit is a read only flag. 0: 1-bit error is not generated 1: 1-bit error is generated Clearing conditions (1) Reset is applied. (2) 1 is written to ECER1C. (3) ECC detection and correction being disabled (ECTHM = 1).
0	ECEMF	ECC Error Message Flag Indicates that the current read data bus contains an error. ECEMF is updated every time RAM data is output. Since the RAM value after reset is undefined, the initial value of this module is determined to be an error, and thus ECEMF may be set. Therefore, the ECEMF value after reset is undefined. 0: The current read data bus contains no bit errors. 1: The current RAM output data contains bit errors. ECEMF remains set as long as RAM data containing a bit error is output with error detection being enabled. Clearing conditions (1) ECC detection and correction being disabled (ECTHM = 1). (2) Decoding circuit input data contains no 1-bit errors.

(2) E710TMC — ECC Test Mode Control Register

E710TMC is a 16-bit register to switch the mode to test mode and controls the mode.

E710TMC can be read and written to using the 16-bit or 8-bit manipulation instruction.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ETMA[1:0]		—	—	—	—	—	—	ECTMCE	—	—	ECTRRS	ECREOS	ECENS	ECDCS	ECREIS
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W*1	R/W*1	R	R	R	R	R	R	R/W	R	R	R/W	R/W	R/W	R/W	R/W

Note 1. This bit is always read as 0.

Table 29.81 E710TMC Register Contents (1/2)

Bit Position	Bit Name	Function
15, 14	ETMA[1:0]	Access Control 1 and 0 to ECC Test Mode Control Enable Bit These bits reserve the write trigger to bit 7. The read value is always 0.
7	ECTMCE	ECC Test Mode Control Enable ECTMCE enables or disables access to the test registers and test control bits. Write access to ETCMCE is enabled when the value of bits 15 and 14 is 10_B. 0: Disables access to the test registers and test control bits. 1: Enables access to the test registers and test control bits.
4	ECTRRS	ECC RAM Read Test Mode Control Enable ECTRRS enables the read status of RAM to be generated by reading the E710TED register, and also allows the RAM output data to be read out when the E710TRC:ECERDB[7:0] bits and the E710TED register are read. Write access to ECTRRS is enabled only when ECTMCE is 1 (can be set simultaneously). ECTRRS is cleared by writing 0 to ECTMCE (can be cleared synchronously). 0: Disables generation of RAM read status for testing when E710TED is read. 1: Enables generation of RAM read status for testing when E710TED is read. When E710TRC:ECERDB[7:0] and E710TED are read, the values of the RAM output data pin are read out.
3	ECREOS	ECC Redundant Bit Output Data Select ECREOS selects either the ECC encoder output data or the ECERDB register value to be output as the ECC redundant bit output. Write access to ECREOS is enabled only when ECTMCE is 1 (can be set simultaneously). ECREOS is cleared by writing 0 to ECTMCE (can be cleared synchronously). 0: Allows encoding result to be output as the ECC redundant bit output. 1: Allows the E710TRC:ECERDB[6:0] value to be output as the ECC redundant bit output.
2	ECENS	ECC Encoder Input Select ECENS selects either the data value from the peripheral module or the test register value (E710TED:ECEDB[31:0]) as the input signal to be encoded. Write access to ECENS is enabled only when ECTMCE is 1 (can be set simultaneously). ECENS is cleared by writing 0 to ECTMCE (can be cleared synchronously). 0: Allows the RAM write data from the peripheral module to be input as the ECC encoder input data. 1: Allows the E710TED:ECEDB[31:0] value to be input as the ECC encoder input data.

Table 29.81 E710TMC Register Contents (2/2)

Bit Position	Bit Name	Function
1	ECDCS	ECC Decoder Input Select ECDCS selects either the lower 32-bit data value from the RAM or the test register value (E710TED:ECEDB[31:0]) as the lower 32-bit data of the input signal to be decoded. Write access to ECDCS is enabled only when ECTMCE is 1 (can be set simultaneously). ECDCS is cleared by writing 0 to ECTMCE (can be cleared synchronously). 0: Allows the lower 32-bit RAM output data to be input to the data area (32 lower-order bits) to the decoder. 1: Allows the E710TED:ECEDB[31:0] value to be input to the data area to the decoder.
0	ECREIS	ECC Redundant Bit Input Data Select ECREIS selects either the upper 7-bit data value from the RAM (redundant bit area) or the test register value (E710TRC:ECERDB[6:0]) as the upper 7-bit data of the input signal to be decoded. Write access to ECREIS is enabled only when ECTMCE is 1 (can be set simultaneously). ECREIS is cleared by writing 0 to ECTMCE (can be cleared synchronously). 0: Allows the upper 7-bit RAM output data to be input to the ECC redundant bit area to the decoder. 1: Allows the E710TRC:ECERDB[6:0] value to be input to the ECC redundant bit area to the decoder.

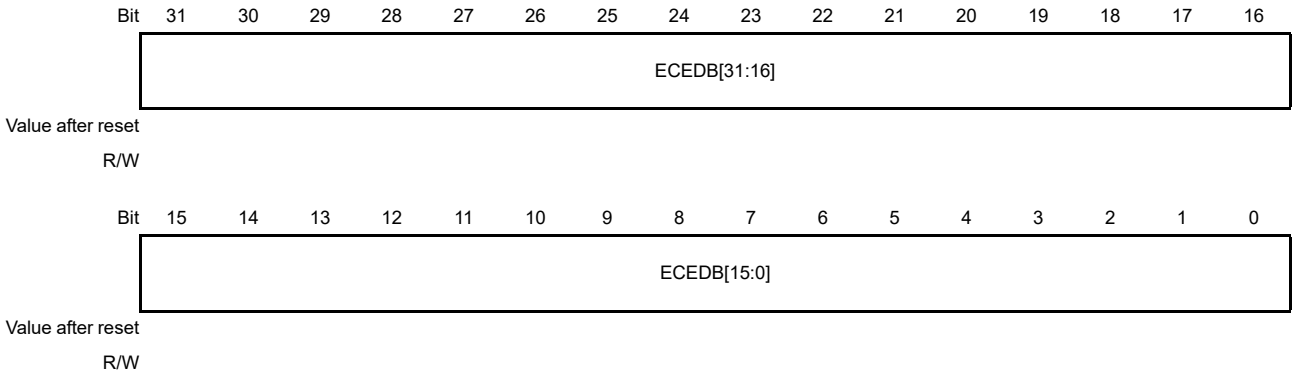
(3) E710TED — ECC Encoder and Decoder Data Test Register

E710TED is a 32-bit data test register for ECC encoding and decoding.

When ECTMCE = 1, E710TED can be read and written to using the 32-bit manipulation instruction.

When ECTMCE = 0, E710TED is always read as 0.

In test mode, the E710TED value can be used as the input data to the encoding circuit and decoding circuit.



NOTE

Changing ECTMCE from 1 to 0 resets E710TED synchronously.

When E710TMC:ECENS = 1, the ECEDB value is input to the encoding circuit and supplied to the RAM.

When E710TMC:ECDCS = 1, the ECEDB value is input as the 31th to 0th bits of the input data to the decoding circuit.

When E710TMC:ECTRRS = 1, reading ECEDB returns the RAM output data instead of the data written to ECEDB.

(4) E710TRC — ECC Redundant Bit Data Control Test Register

E710TRC is a 32-bit test register consisting of four fields (ECSYND, ECHORD, ECECRD, and ECRODM) corresponding to the ECC redundant bit area. Each field can be accessed as the 8-bit register with the same name. For details of each field, refer to the descriptions of these four registers.

When ECTMCE = 0, E710TRC is always read as 0.

When ECTMCE = 1, E710TRC can be read using the 32-bit manipulation instruction.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ECSYND								ECHORD							
Value after reset																
R/W																

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ECECRD								ECERDB							
Value after reset																
R/W																

NOTE

Changing ECTMCE from 1 to 0 resets E710TED synchronously.

(5) ECSYND — ECC Decoder Syndrome Data Register

ECSYND is a read-only register to confirm the syndrome code generated by the decoding circuit in test mode (ECTMCE = 1).

Write access to ECSYND is ignored.

Bit	7	6	5	4	3	2	1	0
	—	SYND[6:0]						
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

When read, the ECSYND bits return the value of the syndrome code (synd[6:0]) generated based on the input data to the decoding circuit.

The ECSYND bits are not holding circuits; the register value changes as the input signal changes.

ECSYND is valid only when ECTMCE = 1, and is always read as 00_H when ECTMCE = 0.

(6) ECHORD — ECC 7-Bit Redundant Data Holding Test Register

ECHORD holds the 7-bit ECC redundant area (upper 7-bit RAM data), which cannot be confirmed by the peripheral module, when the peripheral module accesses the RAM for reading in test mode (ECTMCE = 1).

Bit	7	6	5	4	3	2	1	0
	—	HORD[6:0]						
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

The ECHORD bits are loaded with the upper 7-bit RAM output data at the next rising edge of the operating clock signal after the peripheral module accesses the RAM for reading data in test mode (ECTMCE = 1).

The ECHORD bits are also loaded with the data on the input pins EC7TERI38 to EC7TERI32 at the next operating clock pulse when the ECEDB[15:0] register is read while E710TMC:ECTRRS = 1.

ECHORD is valid only when ECTMCE = 1, and is always read as 00_H when ECTMCE = 0.

(7) ECECRD — ECC Encoder Test Register

ECECRD is a read-only register to read the 7-bit redundant section generated by the encoding circuit in test mode (ECTMCE = 1).

Bit	7	6	5	4	3	2	1	0
	—	ECRD[6:0]						
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

ECECRD is used to confirm the redundant bits generated by the input data from the peripheral module. Here, the read-out data is the result of encoding (ecc[6:0]), not the output value.

ECECRD is valid only when ECTMCE = 1, and is always read as 00_H when ECTMCE = 0.

(8) ECERDB — ECC Redundant Bit Input and Output Substitution Buffer Register

ECERDB is a buffer register for the data that substitutes for the input and output data for the 7-bit ECC redundant data area in test mode (ECTMCE = 1).

ECERDB can be read and written to in ECC test mode (ECTMCE = 1).

Bit	7	6	5	4	3	2	1	0
	—	ERDB[6:0]						
Value after reset	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

When ECREOSS = 1, the ECERDB value, instead of the seven redundant bits generated by the encoding circuit, is output to the pin to be supplied to the RAM.

When ECREIESS = 1, the ECERDB value, instead of the upper seven data bits to be input to the decoding circuit, is handled by the decoding circuit.

When ECTRRS = 1, reading ECERDB returns the signal value supplied to RAM instead of the data written to ECERDB.

(9) E710EAD — ECC Error Address Register

E710EAD is a read-only register to hold the address at which an ECC error has occurred.

If an ECC error is detected while ECC error detection is enabled, the RAM address is latched using the detection signal as a trigger, and the address is stored in E710EAD as the address at which the ECC error has occurred.

The address is stored upon detection of the first ECC error while no error status flag is set. However, if a 1-bit error is followed by a 2-bit error, the address of the latter is stored.

Only one address can be held in E710EAD.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ECEAD[31:16]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ECEAD[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

29.2.9.4 Notification to ECM

Detection of errors in two bits can be set for this module and the ECM is notified of detected errors.

- 1-bit Error Notification
  While EC1EDIC is set to 1_B, when an error is detected in one bit from among bits 0 to 38 of data read from RAM, the ECM is notified of the 1-bit error. When ECER1F or ECER2F is already set, however, the ECM will not be notified of the error.
- 2-bit Error Notification
  While EC2EDIC is set to 1_B, when an error is detected in two bits from among bits 0 to 38 of data read from RAM, the ECM is notified of the 2-bit error. When ECER2F is already set, however, the ECM will not be notified of the error.

29.2.9.5 Test Function

(1) Writing to RAM Data

Write data to the peripheral RAM. However, the ECC corresponding to the written data will be written to the ECC bits simultaneously. In order to write a specified value to the ECC bits, use the ECC test mode described in (3) below.

(2) Reading RAM Data

(a) Set the ECTHM bit in the ECC control register to 1 to disable ECC error detection and correction.

(b) Read the peripheral RAM. Since neither error detection nor correction proceeds when the peripheral RAM is read, the RAM data is read unchanged.

How to exit this test mode:

(a) Set the ECTHM bit in the ECC control register to 0 to enable ECC error detection and correction.

(3) Writing to the ECC Bits

(a) Set the ECTMCE bit in the ECC test mode control register to 1 to set ECC test mode.

(b) Write the value for writing to the ECC bits to the E710TRC.ECERDB[6:0] bits.

(c) Set the ECREOS bit in the ECC test mode control register to 1 to select writing of the value of the E710TRC.ECERDB[6:0] bits to the ECC bits.

(d) When data are written to the peripheral RAM, the value in the E710TRC.ECERDB[6:0] bits will be written to the ECC bits.

How to exit this test mode:

(a) Set the ECTMCE bit in the ECC test mode control register to 0 to set normal mode.

(4) Reading the ECC Bits

(a) Set the ECTMCE bit in the ECC test mode control register to 1 to set ECC test mode.

(b) When data in the peripheral RAM is read, the ECC bits are stored in the E710TRC.ECHORD[6:0] bits.

How to exit this test mode:

(a) Set the ECTMCE bit in the ECC test mode control register to 0 to set normal mode.

29.2.10 ECC for Peripheral RAM (16 Bits)

29.2.10.1 Overview

This is an ECC module for the RAM of the following peripheral module.

DFE

Error Detection and Correction

Six-bit ECC data is appended to the 16-bit RAM data.

This ECC module provides 2-bit ECC error detection and 1-bit ECC error detection and correction.

CAUTION

If three or more bits of errors occur, they cannot be detected by this module. In this case, such errors may be detected as 1 or 2-bit errors; no errors may be detected; or a non-inverted bit may be corrected depending on the setting.

Enabling or Disabling ECC Error Detection and Correction

- ECC error detection can be either enabled or disabled.
- One-bit ECC error correction can be either enabled or disabled.
- If all the bits of RAM output data are stuck to 0 or 1, it is detected as a 2-bit ECC error.

Error Notification

- The ECM is notified when 2-bit ECC errors are detected (this can be enabled or disabled).
- The ECM is notified when 1-bit ECC errors are detected (this can be enabled or disabled).

Once the ECM has been notified of an error, even if another ECC error is detected, the ECM is not notified until the error status bit corresponding to the initial error is cleared.

Error Status

- Detection of 2- and 1-bit ECC errors can be monitored.
- Special registers are provided to clear error status.

Address Capture

- Only one address at which an ECC error has occurred can be captured.
- When a 2-bit or 1-bit ECC error is detected, the address where the error was found is captured (when the first (1-bit or 2-bit) error is detected after the flag is cleared).

Testing Function (Error Insertion)

- By setting the test mode, register values can be used as the data to be output to the RAM. When a peripheral module writes to the RAM, the ECEDB[15:0] register value can be written to the RAM data section, and the ECERDB[5:0] register value can be written to the ECC redundant bit section.
- By setting the test mode, the ECC redundant bit section can be latched when RAM data is read, and the value can be confirmed.
- By setting the test mode, the ECC redundant bit (encoding circuit) and syndrome code (decoding circuit), which are generated from the input data, can be confirmed.

29.2.10.2 List of Registers

(1) List of ECC Modules

The RAMs of the multiple peripheral functions are provided with the ECC modules. The following table shows the peripheral functions provided with the ECC modules, the corresponding ECC module names, and base addresses of the ECC modules.

Table 29.82 Peripheral Functions, Corresponding ECC Module Names, and Base Addresses of ECC Modules

Peripheral Functions		ECC Module Names and Register Base Addresses			
		Master Side*1		Checker Side*1	
		Module Name	Base Address <base_addr>	Module Names	Base Address <base_addr>
DFE	Data memory (DMEM0)	E6DF0M	FFC7 D400$_H$	E6DF0C	FFC7 D600$_H$
	Data memory (DMEM1)	E6DF1M	FFC7 D800$_H$	E6DF1C	FFC7 DA00$_H$

Note 1. Two ECC modules are provided to support BIST, one for the master and the other for the checker. For details, refer to **Section 29.7, BIST**.

(2) List of Registers

Each ECC module has the registers shown in the following table.

Table 29.83 List of ECC Module Registers

Register Name	Symbol	R/W	Value after Reset	Address	Access Size
ECC control register*1	E610CTL	R/W	001X$_H$	<base_addr> + 00$_H$	16/8
ECC test mode control register	E610TMC	R/W	0000$_H$	<base_addr> + 04$_H$	16/8
ECC redundant bit data control test register	E610TRC	R	0000 0000$_H$	<base_addr> + 08$_H$	32
ECC encoder and decoder data test register	E610TED	R/W	0000 0000$_H$	<base_addr> + 0C$_H$	32
ECC error address register	E610EAD	R	0000 0000$_H$	<base_addr> + 10$_H$	32

Note 1. The reset value of the LSB in the ECC control register is undefined.

(3) Register Map

Table 29.84 List of Register Map

Abbreviation	31 24	23 16	15 8	7 0	Address
E610CTL	— (00$_H$)	— (00$_H$)	ECCTL[15:8]	ECCTL[7:0]	nn00$_H$
E610TMC	— (00$_H$)	— (00$_H$)	ECTMC[15:8]	ECTMC[7:0]	nn04$_H$
E610TRC	ECSYND[7:0]	ECHORD[7:0]	ECECRD[7:0]	ECRODM[7:0]	nn08$_H$
E610TED	ECEDB[31:24]	ECEDB[23:16]	ECRIDM[15:8]	ECERDB[7:0]	nn0C$_H$
E610EAD	ECEAD[31:24]	ECEAD[23:16]	ECEAD[15:8]	ECEAD[7:0]	nn10$_H$

29.2.10.3 Register Details

(1) E610CTL — ECC Control Register

E610CTL controls the status and modes of the ECC module.

E610CTL can be read and written to using the 16-bit or 8-bit manipulation instruction.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	EMCA[1:0]		—	—	—	ECER2C	ECER1C	—	ECTHM	—	EC1ECP	EC2EDIC	EC1EDIC	ECER2F	ECER1F	ECEMF
Value after reset	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	Un-defined
R/W	R/W*1	R/W*1	R	R	R	R/W*1	R/W*1	R	R/W	R	R/W	R/W	R/W	R	R	R

Note 1. This bit is always read as 0.

Table 29.85 E610CTL Register Contents (1/2)

Bit Position	Bit Name	Function
15, 14	EMCA[1:0]	Access Control 1 and 0 to ECC Mode Select Bit These bits reserve the write trigger to bit 7. The read value is always 0.
10	ECER2C	2-Bit ECC Error Detection Flag Clear Clears the bit 2 (ECER2F) status flag. The read value is always 0, and writing 0 to ECER2C does not change the state. If writing 1 to ECER2C conflicts with the condition for setting bit 2, the former takes priority. Writing 1 to ECER2C while ECER2F is set clears ECER2F.
9	ECER1C	1-Bit ECC Error Detection And Correction Accumulated Flag Clear Clears the bit 1 (ECER1F) status flag. The read value is always 0, and writing 0 to ECER1C does not change the state. If writing 1 to ECER1C conflicts with the condition for setting bit 1, the former takes priority. Writing 1 to ECER1C while ECER1F is set clears ECER1F.
7	ECTHM	ECC Function Disable Selects the ECC decoding operation. Write access to ECTHM is enabled when the value of bits 15 and 14 is 01_B. Therefore, only the 16-bit manipulation instruction is valid. Setting ECTHM to 1 disables error detection and bit correction. Here, if the data to be output to the peripheral module contains an error, the data is output without bit correction. Setting ECTHM to 1 has no effect on the encoder side. 0: Enables ECC detection and correction 1: Disables ECC detection and correction. Has no effect on the encoder side.
5	EC1ECP	1-Bit ECC Error Correction Enable 0: Enables 1-bit error correction upon error detection. 1: Disables 1-bit error correction upon error detection.
4	EC2EDIC	2-Bit ECC Error Detection Notification Enable 0: When a 2-bit error is detected, the ECM is not notified of the error. 1: When a 2-bit error is detected, the ECM is notified of the error.
3	EC1EDIC	1-Bit ECC Error Detection Notification Enable 0: When a 1-bit error is detected, the ECM is not notified of the error. 1: When a 1-bit error is detected, the ECM is notified of the error.
2	ECER2F	2-Bit ECC Error Detection Flag Indicates that errors have been detected at two bits in bits 0 to 21 of the data read from the RAM during RAM read access while error detection is enabled. This bit is a read only flag. 0: 2-bit error is not generated. 1: 2-bit error is generated Clearing conditions (1) Reset is applied. (2) 1 is written to ECER2C. (3) ECC detection and correction being disabled (ECTHM = 1).

Table 29.85 E610CTL Register Contents (2/2)

Bit Position	Bit Name	Function
1	ECER1F	1-Bit ECC Error Detection And Correction Flag Indicates that an error has been detected at one bit in bits 0 to 21 of the data read from the RAM during RAM read access while error detection is enabled. This bit is a read only flag. 0: 1-bit error is not generated. 1: 1-bit error is generated Clearing conditions (1) Reset is applied. (2) 1 is written to ECER1C. (3) ECC detection and correction being disabled (ECTHM = 1).
0	ECEMF	ECC Error Message Flag Indicates that the current read data bus contains an error. Since the RAM value after reset is undefined, the initial value of this module is determined to be an error, and thus ECEMF may be set. Therefore, the ECEMF value after reset is undefined. 0: The current read data bus contains no bit errors. 1: The current RAM output data contains bit errors. ECEMF remains set as long as RAM data containing a bit error is output with error detection being enabled. Clearing conditions (1)ECC detection and correction being disabled (ECTHM = 1). (2) Decoding circuit input data contains no 1-bit errors.

(2) E610TMC — ECC Test Mode Control Register

E610TMC is a 16-bit register to switch the mode to test mode and controls the mode.

E610TMC can be read and written to using the 16-bit or 8-bit manipulation instruction.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ETMA[1:0]		—	—	—	—	—	—	ECTMCE	—	—	ECTRRS	ECREOS	ECENS	ECDCS	ECREIS
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W*1	R/W*1	R	R	R	R	R	R	R/W	R	R	R/W	R/W	R/W	R/W	R/W

Note 1. This bit is always read as 0.

Table 29.86 E610TMC Register Contents (1/2)

Bit Position	Bit Name	Function
15, 14	ETMA[1:0]	Access Control 1 and 0 to ECC Test Mode Control Enable Bit These bits reserve the write trigger to bit 7. The read value is always 0.
7	ECTMCE	ECC Test Mode Control Enable ECTMCE enables or disables access to the test registers and test control bits. Write access to ETCMCE is enabled when the value of bits 15 and 14 is 10_B. 0: Disables access to the test registers and test control bits. 1: Enables access to the test registers and test control bits.
4	ECTRRS	ECC RAM Read Test Mode Control Enable ECTRRS enables the read status of RAM to be generated by reading the E610TED register, and also allows the RAM output data to be read out when the E610TED:ECEDB[15:0] and E610TED:ECRIDM[7:0] are read. Write access to ECTRRS is enabled only when ECTMCE is 1 (can be set simultaneously). ECTRRS is cleared by writing 0 to ECTMCE (can be cleared synchronously). 0: Disables generation of RAM read status for testing when E610TED is read. 1: Enables generation of RAM read status for testing when E610TED is read. When E610TED:ECEDB[15:0] and E610TED:ECRIDM[7:0] are read, the values of the RAM output data pin are read out.
3	ECREOS	ECC Redundant Bit Output Data Select ECREOS selects either the ECC encoder output data or the ECERDB register value to be output as the ECC redundant bit output. Write access to ECREOS is enabled only when ECTMCE is 1 (can be set simultaneously). ECREOS is cleared by writing 0 to ECTMCE (can be cleared synchronously). 0: Allows encoding result to be output as the ECC redundant bit output. 1: Allows the E610TED:ECERDB[5:0] value to be output as the ECC redundant bit output.
2	ECENS	ECC Encoder Input Select ECENS selects either the data value from the peripheral module or the test register value (E610TED: ECEDB[15:0]) as the input signal to be encoded. Write access to ECENS is enabled only when ECTMCE is 1 (can be set simultaneously). ECENS is cleared by writing 0 to ECTMCE (can be cleared synchronously). 0: Allows the RAM write data from the peripheral module to be input as the ECC encoder input data. 1: Allows the E610TED:ECEDB[15:0] value to be input as the ECC encoder input data.

Table 29.86 E610TMC Register Contents (2/2)

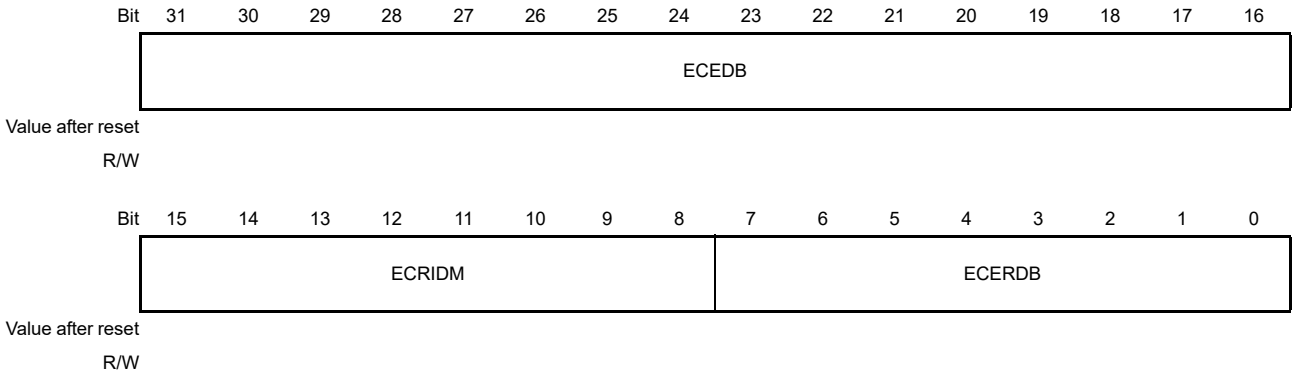
Bit Position	Bit Name	Function
1	ECDCS	ECC Decoder Input Select ECDCS selects either the lower 16-bit data value from the RAM or the test register value (E610TED:ECEDB[15:0]) as the lower 32-bit data of the input signal to be decoded. Write access to ECDCS is enabled only when ECTMCE is 1 (can be set simultaneously). ECDCS is cleared by writing 0 to ECTMCE (can be cleared synchronously). 0: Allows the lower 16-bit RAM output data to be input to the data area (lower 16 bits) to the decoder. 1: Allows the E610TED:ECEDB[15:0] value to be input to the data area to the decoder.
0	ECREIS	ECC Redundant Bit Input Data Select ECREIS selects either the upper 6-bit data value from the RAM (redundant bit area) or the test register value (E610TED:ECERDB[5:0]) as the upper 6-bit data of the input signal to be decoded. Write access to ECREIS is enabled only when ECTMCE is 1 (can be set simultaneously). ECREIS is cleared by writing 0 to ECTMCE (can be cleared synchronously). 0: Allows the upper 6-bit RAM output data to be input to the ECC redundant bit area to the decoder. 1: Allows the E610TED:ECERDB[5:0] value to be input to the ECC redundant bit area to the decoder.

(3) E610TED — ECC Encoder and Decoder Data Test Register

E610TED is a 32-bit test register consisting of three fields (ECEDB, ECRIDM, and ECERDB) corresponding to the ECC data area and redundant bit area. Each field can be accessed as the register with the same name. For details of each field, refer to the descriptions of these three registers.

When ECTMCE = 1, E610TED can be read and written to using the 32-bit manipulation instruction.

When ECTMCE = 0, E610TED is always read as 0.



NOTE

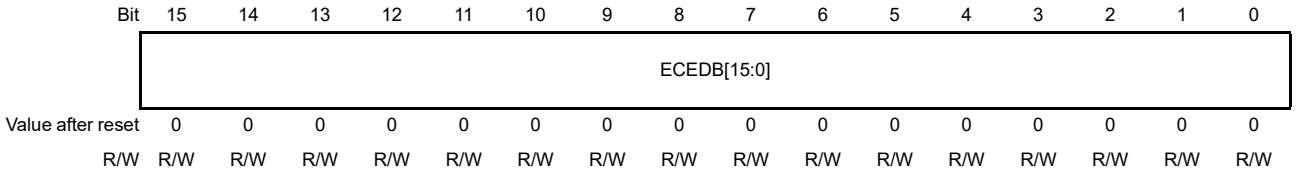
Changing ECTMCE from 1 to 0 resets E710TED synchronously.

(4) ECEDB — ECC Encoder and Decoder Data Input and Output Substitution Buffer Register

ECEDB is a 16-bit data buffer register for ECC encoding and decoding.

ECEDB can be read and written to in test mode (ECTMCE = 1).

When ECTMCE = 1, the ECEDB value can be used as the input data to the encoding circuit and decoding circuit.



When E610TMC:ECENS = 1, the ECEDB value is input to the encoding circuit and supplied to the RAM.

When E610TMC:ECDCS = 1, the ECEDB value is input as the 15th to 0th bits of the input data to the decoding circuit. When E610TMC:ECTRRS = 1, reading ECEDB returns the RAM output data instead of the data written to ECEDB.

(5) ECRIDM — ECC Redundant Bit Input Data Monitoring Register

ECRIDM is a read-only register to monitor the upper 6-bit data (redundant bit area) value from the RAM in test mode (ECTMCE = 1) when E610TMC:ECTRRS = 1. With ECRIDM, the RAM output data can be confirmed easily.

When ECTRRS = 0, ECRIDM is always read as 0.

Bit	7	6	5	4	3	2	1	0
	—	—	RIDM[5:0]					
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

(6) ECERDB — ECC Redundant Bit Input and Output Substitution Buffer Register

ECERDB is a buffer register for the data that substitutes for the input and output data for the 6-bit ECC redundant data area in test mode (ECTMCE = 1).

ECERDB can be read and written to in test mode (ECTMCE = 1).

Bit	7	6	5	4	3	2	1	0
	—	—	ERDB[5:0]					
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W

When ECREOS = 1, the ECERDB value, instead of the six redundant bits generated by the encoding circuit, is output to the ECC redundant bit output pin to be supplied to the RAM.

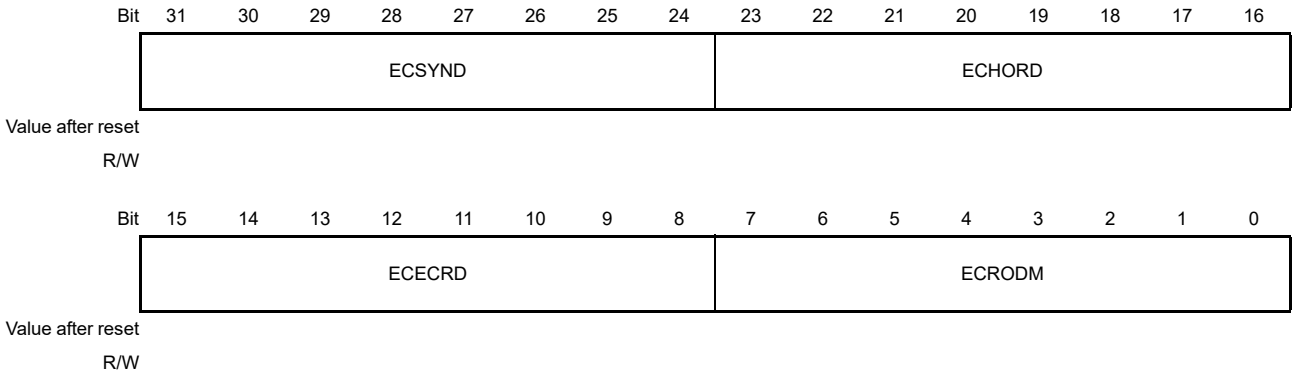
When ECREIS = 1, the ECERDB value, instead of the upper six data bits to be input to the decoding circuit, is handled by the decoding circuit.

(7) E610TRC — ECC Redundant Bit Data Control Test Register

E610TRC is a 32-bit test register consisting of four fields (ECSYND, ECHORD, ECECRD, and ECRODM) corresponding to the ECC redundant bit area. Each field can be accessed as the 8-bit register with the same name. For details of each field, refer to the descriptions of these four registers.

When ECTMCE = 0, E610TRC is always read as 0.

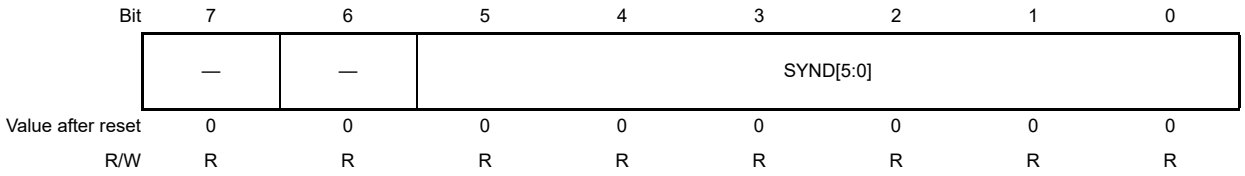
When ECTMCE = 1, E610TRC can be read using the 32-bit manipulation instruction.



(8) ECSYND — ECC Decoder Syndrome Data Register

ECSYND is a read-only register to confirm the syndrome code generated by the decoding circuit in test mode (ECTMCE = 1).

Write access to ECSYND is ignored.



When read, the ECSYND bits return the value of the syndrome code (synd[5:0]) generated based on the input data to the decoding circuit.

The ECSYND bits are not holding circuits; the register value changes as the input signal changes.

ECSYND is valid only when ECTMCE = 1, and is always read as 00_H when ECTMCE = 0.

(9) ECHORD — ECC 6-Bit Redundant Data Holding Test Register

ECHORD holds the 6-bit ECC redundant area (upper 6-bit RAM data), which cannot be confirmed by the peripheral module, when the peripheral module accesses the RAM for reading in test mode (ECTMCE = 1).

Bit	7	6	5	4	3	2	1	0
	—	—	HORD[5:0]					
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

The ECHORD bits are loaded with the upper 6-bit RAM output data at the next rising edge of the operating clock signal after the peripheral module accesses the RAM for reading data in test mode (ECTMCE = 1).

The ECHORD bits are also loaded with the data on the input pins at the next operating clock pulse when the ECEDB[15:0] register is read while E610TMC:ECTRRS = 1.

ECHORD is valid only when ECTMCE = 1, and is always read as 00_H when ECTMCE = 0.

(10) ECECRD — ECC Encoder Test Register

ECECRD is a read-only register to read the 6-bit redundant section generated by the encoding circuit in test mode (ECTMCE = 1).

Bit	7	6	5	4	3	2	1	0
	—	—	ECRD[5:0]					
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

ECECRD is used to confirm the redundant bits generated by the input data from the peripheral module. Here, the read-out data is the result of encoding (ecc[5:0]), not the value output from the ECC redundant bits.

ECECRD is valid only when ECTMCE = 1, and is always read as 00_H when ECTMCE = 0.

(11) ECRODM — ECC Redundant Bit Output Data Monitoring Register

ECRODM is a read-only register to monitor the value of the ECC redundant bit output pin in test mode (ECTMCE = 1).

With ECRODM, the upper 6-bit output data to the RAM can be confirmed easily.

Bit	7	6	5	4	3	2	1	0
	—	—	RODM[5:0]					
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

ECRODM is valid only when ECTMCE = 1, and is always read as 00_H when ECTMCE = 0.

(12) E610EAD — ECC Error Address Register

E610EAD is a read-only register to hold the address at which an ECC error has occurred.

If an ECC error is detected while ECC error detection is enabled, the RAM address is latched using the detection signal as a trigger, and the address is stored in E610EAD as the address at which the ECC error has occurred.

The address is stored upon detection of the first ECC error while no error status flag is set. However, if a 1-bit error is followed by a 2-bit error, the address of the latter is stored.

Only one address can be held in E610EAD.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ECEAD[31:16]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ECEAD[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

29.2.10.4 Notification to ECM

Detection of errors in two bits can be set for this module and the ECM is notified of detected errors.

- 1-bit Error Notification
  While EC1EDIC is set to 1_B, when an error is detected in one bit from among bits 0 to 21 of data read from RAM, the ECM is notified of the 1-bit error. When ECER1F or ECER2F is already set, however, the ECM will not be notified of the error.
- 2-bit Error Notification
  While EC2EDIC is set to 1_B, when an error is detected in two bits from among bits 0 to 21 of data read from RAM, the ECM is notified of the 2-bit error. When ECER2F is already set, however, the ECM will not be notified of the error.

29.2.10.5 Test Function

(1) Writing to RAM Data

Write data to the peripheral RAM. However, the ECC corresponding to the written data will be written to the ECC bits simultaneously. In order to write a specified value to the ECC bits, use the ECC test mode described in (3) below.

(2) Reading RAM Data

(a) Set the ECTHM bit in the ECC control register to 1 to disable ECC error detection and correction.

(b) Read the peripheral RAM. Since neither error detection nor correction proceeds when the peripheral RAM is read, the RAM data is read unchanged.

How to exit this test mode:

(a) Set the ECTHM bit in the ECC control register to 0 to enable ECC error detection and correction.

(3) Writing to the ECC Bits

(a) Set the ECTMCE bit in the ECC test mode control register to 1 to set ECC test mode.

(b) Write the value for writing to the ECC bits to the E610TED.ECERDB[5:0] bits.

(c) Set the ECREOS bit in the ECC test mode control register to 1 to select writing of the value of the E610TED.ECERDB[5:0] bits to the ECC bits.

(d) When data are written to the peripheral RAM, the value in the E610TED.ECERDB[5:0] bits will be written to the ECC bits.

How to exit this test mode:

(a) Set the ECTMCE bit in the ECC test mode control register to 0 to set normal mode.

(4) Reading the ECC Bits

(a) Set the ECTMCE bit in the ECC test mode control register to 1 to set ECC test mode.

(b) When data in the peripheral RAM is read, the ECC bits are stored in the E610TRC.ECHORD[5:0] bits.

How to exit this test mode:

(a) Set the ECTMCE bit in the ECC test mode control register to 0 to set normal mode.

29.2.11 Data Parity for Data Transfer Paths

The following shows the transfer paths to which data parity is applied. Data parity errors can be detected during data transfer from the access sources to destinations shown below. If a parity error is detected in any of the paths, the ECM is notified of the error.

Table 29.87 Transfer Paths to which Data Parity is Applied

Access Source (Master)	Access Destination (Slave)
CPU1, PCU, DMAC, DTS	INTC2, DMA, DTS, APA, ATU-IV, CSIH, ΔΣAD, ADC port group 0, DNF, edge detection DNF
DMAC, DTS	CPU1 local RAM, global RAM

When a data parity error occurs during read operation, 0 is returned to CPU1 and error data is returned to other masters.

When a data parity error occurs during write operation, the write operation is cancelled if the slave is INTC 2, APA, ATU-IV, ΔΣAD, ADC (except IFC), DNF, or edge detection DNF.

Some of the parity-applied modules have the control registers in the parity controllers (encoder and decoder), which can enable or disable parity detection and hold status upon detection of an error. If a parity error is detected in these modules, it is possible to identify the access that caused the error.

On the other hand, parity controllers which do not have control registers always perform parity detection. Though the status upon detection of an error is not retained in these controllers, whether an error has been detected or not is retained in the ECM.

29.2.11.1 List of Registers

Table 29.88 List of Registers

Address	Symbol	Register Name	R/W	Value after Reset	Access Size
<Base_addr>+0_H	APDPERRST_xx	P-Bus data parity status register xx	R	0000 0000_H	8/16/32
<Base_addr>+4_H	APDPERRSTC_xx	P-Bus data parity status clear register xx	W	0000 0000_H	8/16/32
<Base_addr>+8_H	APDPTMC_xx	P-Bus data parity test mode control register xx	R/W	0000 0000_H	16/32
<Base_addr>+C_H	APDPERRADR_xx	P-Bus data parity error address register xx	R	0000 0000_H	32

The symbol xx in the above table indicates a parity-applied module for which the control register is provided. **Table 29.89** shows the parity-applied modules and base addresses<base_addr>.

Table 29.89 List of Data Parity Control Modules (1/2)

Module to Which Parity is Applied	xx	<base_addr>
INTC2	INTC2	FFC6 8800_H
DMA_DTS	PDMA	FFC6 8900_H
APA (other than APA input selector)	APAA	FFDC 2000_H
APA (APA input selector)	APGA	FFDC 2020_H
ATU-IV	ATU4	FFDD $F000_H$
CSIH0 (group A)*1	CS0A	FFF9 6000_H

Table 29.89 List of Data Parity Control Modules (2/2)

Module to Which Parity is Applied	xx	<base_addr>
CSIH0 (group B)*1	CS0B	FFF9 6020_H
CSIH1 (group A)*1	CS1A	FFF9 6040_H
CSIH1 (group B)*1	CS1B	FFF9 6060_H
CSIH2 (group A)*1	CS2A	FFF9 6080_H
CSIH2 (group B)*1	CS2B	FFF9 $60A0_H$
CSIH3 (group A)*1	CS3A	FFF9 $60C0_H$
CSIH3 (group B)*1	CS3B	FFF9 $60E0_H$
ΔΣAD (shared register)	DADC	FFC8 7000_H
ΔΣAD (CH0 register)	DAD0	FFC8 7020_H
ΔΣAD (CH1 register)	DAD1	FFC8 7040_H
ΔΣAD (CH2 register)	DAD2	FFC8 7060_H
ΔΣAD (CH3 register)	DAD3	FFC8 7080_H
ΔΣAD (CH4 register)	DAD4	FFC8 $70A0_H$
ΔΣAD (CH5 register)	DAD5	FFC8 $70C0_H$
ΔΣAD (CH6 register)	DAD6	FFC8 $70E0_H$
ΔΣAD (CH7 register)	DAD7	FFC8 7100_H
ADC0	ADC0	FFC8 8000_H
ADC1	ADC1	FFC8 8020_H
ADC (IFC)	IFC	FFC8 8100_H
ADC (ASF)	ASF	FFC8 8120_H
Port group 0 (group A)*2	PT0A	FFC8 5000_H
Port group 0 (group B)*2	PT0B	FFC8 5020_H
Port group n (group A)*2	PT1A	FFC8 5040_H
Port group n (group B)*2	PT1B	FFC8 5060_H
DNF0	DNF0	FFC8 $50C0_H$
DNF1	DNF1	FFC8 $50E0_H$
Edge detection DNF2	DNF2	FFC8 5800_H
Edge detection DNF3	DNF3	FFC8 5820_H
Edge detection DNF4	DNF4	FFC8 5840_H
Edge detection DNF5	DNF5	FFC8 5860_H
Edge detection DNF6	DNF6	FFC8 5880_H
Edge detection DNF7	DNF7	FFC8 $58A0_H$

Note 1. The CSIHx registers are divided into the following two groups and controlled separately.
Group A: CSIHnCTL0 to CSIHnCTL2, CSIHnSTR0, and CSIHnSTCR0
Group B: Other than above.

Note 2. The port group registers are divided into the following two groups and controlled separately.
Group A: Registers other than PCRn_m registers (see **Section 2.1.4.1 Table 2.6**)
Group B: PCRn_m
Here, n represents the port group number (n = 0 to 17), and m represents the bit numbers (m = 0 to 15) in each port group.

29.2.11.2 Details of Registers

(1) APDPERRST_xx — P-Bus Data Parity Status Register

For the symbol xx, see **Table 29.89, List of Data Parity Control Modules**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	APDPERR
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 29.90 APDPERRST Register Contents

Bit Position	Bit Name	Function
31 to 1	—	Reserved These bits are always read as 0. The write value should also be 0.
0	APDPERR	Data Parity Error Monitor Flag Indicates occurrence of the parity error. APDPERR is cleared by setting the Data Parity error monitor flag clear bit (APDPERRC) to 1. 0: Indicates that a parity error has not occurred. 1: Indicates that a parity error has occurred.

(2) APDPERRSTC_xx — P-Bus Data Parity Status Clear Register

For the symbol xx, see **Table 29.89, List of Data Parity Control Modules**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	APDPERRC
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	W

Table 29.91 APDPERRSTC Register Contents

Bit Position	Bit Name	Function
31 to 1	—	Reserved These bits are always read as 0. The write value should also be 0.
0	APDPERRC	Clears the Data Parity error monitor flag (APDPERR). Writing 1 to this bit with the APDPERR bit set clears APDPERR. This bit is always read as 0.

(3) APDPTMC_xx — P-Bus Data Parity Test Mode Control Register

For the symbol xx, see **Table 29.89, List of Data Parity Control Modules**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	APDPTMC[1:0]		—	—	—	—	—	—	—	—	—	—	APDPEIC3	APDPEIC2	APDPEIC1	APDPEIC0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Table 29.92 APDPTMC Register Contents

Bit Position	Bit Name	Function
31 to 16	—	Reserved These bits are always read as 0. The write value should also be 0.
15, 14	APDPTMC[1:0]	Data Parity Test Mode Control Setting APDPTMC[1:0] to 01_B enables write-accesses to the error insertion control bits 3 to 0 (APDPEIC3 to APDPEIC0). When APDPTMC[1:0] are not 01_B, APDPEIC3 to APDPEIC0 are not written to even though a write-access is attempted. These bits are always read as 0.
13 to 4	—	Reserved These bits are always read as 0. The write value should also be 0.
3	APDPEIC3	Byte-Lane 3 Error Insertion Control Changes the type of the parity bit generation and check for byte-lane 3 (bits 31 to 24) to odd parity. APDPEIC3 can be written to when APDPTMC[1:0] = 01_B. 0: Parity bit generation and check is performed with even parity. 1: Parity bit generation and check is performed with odd parity. (Error insertion)
2	APDPEIC2	Byte-Lane 2 Error Insertion Control Changes the type of the parity bit generation and check for byte-lane 2 (bits 23 to 16) to odd parity. APDPEIC2 can be written to when APDPTMC[1:0] = 01_B. 0: Parity bit generation and check is performed with even parity. 1: Parity bit generation and check is performed with odd parity. (Error insertion)
1	APDPEIC1	Byte-Lane 1 Error Insertion Control Changes the type of the parity bit generation and check for byte-lane 1 (bits 15 to 8) to odd parity. APDPEIC1 can be written to when APDPTMC[1:0] = 01_B. 0: Parity bit generation and check is performed with even parity. 1: Parity bit generation and check is performed with odd parity. (Error insertion)
0	APDPEIC0	Byte-Lane 0 Error Insertion Control Changes the type of the parity bit generation and check for byte-lane 0 (bits 7 to 0) to odd parity. APDPEIC0 can be written to when APDPTMC[1:0] = 01_B. 0: Parity bit generation and check is performed with even parity. 1: Parity bit generation and check is performed with odd parity. (Error insertion)

(4) APDPERRADR_xx — P-Bus Data Parity Error Address Register

For the symbol xx, see **Table 29.89, List of Data Parity Control Modules**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	APDPERRADR[31:16]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	APDPERRADR[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 29.93 APDPERRADR Register Contents

Bit Position	Bit Name	Function
31 to 0	APDPERRADR [31:0]	Data Parity Error Address Holds the address at which the first parity error has occurred while the Data Parity error monitor flag (APDPAERR) is not set. The retained information is not updated even though the next parity error occurs while DPAERR is set.

29.3 Lockstep

This product incorporates the CPU1 with the lockstep function to quickly detect CPU failures without special software. The CPU1 executes the program using two different cores, that is, master core and checker core, and constantly compares the execution results of the two cores. When the results do not agree, the CPU1 determines that an error has occurred in one of the cores, and notifies a lockstep compare error to the ECM.

Bus outputs to be compared are outputs to the local RAM for CPU1, global RAM, a CPU peripheral, interconnect, the P-bus, code flash memory, and the tag RAM and data RAM of the instruction cache.

The lockstep function of the CPU1 features failure insertion, with which errors can be intentionally caused and thus self-diagnosis of the lockstep operation is possible.

29.3.1 List of Registers

Table 29.94 List of Registers

Address	Symbol	Register Name	R/W	Value after Reset	Access Size
FFFE ED00$_H$	TESTCOMPREG0	Comparator test register 0	R/W	0000 0000$_H$	8/16/32
FFFE ED04$_H$	TESTCOMPREG1	Comparator test register 1	R/W	0000 0000$_H$	8/16/32

These registers are placed in the CPU Peripheral of the CPU1. These registers can only be accessed by the CPU1.

29.3.2 Details of Registers

29.3.2.1 TESTCOMPREG0 — Comparator Test Register 0

TESTCOMPREG0 is test register 0 used for the lockstep function of the CPU1.

Combining TESTCOMPREG0 with TESTCOMPREG1 enables self-diagnosis of the lockstep function. The following gives an example of self-diagnosis procedure.

(1) Write arbitrary value to TESTCOMPREG0.

(2) Write a different value to TESTCOMPREG1.

(3) Read TESTCOMPREG0. The different values are read and sent to the master core and checker core.

(4) Using the values read, run the comparator to be diagnosed.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TESTCOMPREG0[31:16]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TESTCOMPREG0[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 29.95 TESTCOMPREG0 Register Contents

Bit Position	Bit Name	Function
31 to 0	TESTCOMPREG0[31:0]	Write: Data is written to each byte. Read: PE1: TESTCOMPREG0[31:0] value is read. PE1C: TESTCOMPREG1[31:0] value is read.

29.3.2.2 TESTCOMPREG1 — Comparator Test Register 1

TESTCOMPREG1 is test register 0 used for the lockstep function of the CPU1.

Combining TESTCOMPREG1 with TESTCOMPREG0 enables self-diagnosis of the lockstep function.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TESTCOMPREG1[31:16]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TESTCOMPREG1[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 29.96 TESTCOMPREG1 Register Contents

Bit Position	Bit Name	Function
31 to 0	TESTCOMPREG1[31:0]	Write: Data is written to each byte. Read: PE1: TESTCOMPREG1[31:0] value is read. PE1C: TESTCOMPREG0[31:0] value is read.

29.4 Memory Protection

29.4.1 Overview

This product incorporates the memory protection function to prevent erroneous accesses to data in memories and control registers of the peripheral circuits.

- MPU
  The CPU1 and PCU protect memories against illegal accesses from the CPU1 and PCU themselves. Accesses to addresses that are prohibited by the MPU are never issued by the CPU1 or PCU. For details, refer to RH850G3M User's Manual: Software and RH850G3K User's Manual: Software.
- Slave Guard
  A specific memory is protected against illegal accesses from any bus master. Slave guard includes the following guard types. The details of each type are given in the following sections.

  – PEG
    The local RAM is protected against illegal accesses. However, accesses from the CPU incorporating the local RAM itself are excluded. For example, accesses from the CPU1 to local RAM in the CPU1 are not rejected by the PEG.
    For details, refer to **Section 3, CPU System**.
  – IPG
    The CPU Peripheral is protected against illegal accesses.
    For details, refer to **Section 3, CPU System**.
  – GRG
    The global RAM is protected against illegal accesses.
  – PBG
    The control registers in the peripheral circuits and memories are protected against illegal accesses. For details, refer to **Section 29.4.3**

29.4.1.1 Identifiers for Slave Guard

For the slave guard function, the type of illegal accesses to be rejected can be designated using the following identifiers.

Table 29.97 Identifiers for Slave Guard

Identifier	Function
UM	When the CPU makes an access, the operating mode of the CPU is indicated. 0: Supervisor mode 1: User mode When the DMAC or DTS makes an access, the value of this identifier is the value in the channel master setting register. When the other master makes an access, the value of this identifier is always 0.
SPID	When the CPU makes an access, the system protection identifier SPID that is assigned to the CPU is indicated. When the DMAC or DTS makes an access, the value of this identifier is the value in the channel master setting register. When the other master makes an access, the value of this identifier is always 00_B.
PEID	The access source bus master is indicated. 000_B: Reserved 001_B: CPU1 010_B: Reserved 011_B: PCU 100_B: Other bus master 101_B: Reserved 110_B: Reserved 111_B: Reserved When the DMAC or DTS makes an access, the value of this identifier is the value in the channel master setting register.

29.4.2 GRG (Global RAM Guard)

This product is provided with 4-channel GRG, which is described in detail in the following sections.

29.4.2.1 List of Registers

Table 29.98 List of Registers

Address	Symbol	Register Name	R/W	Value after Reset	Access Size
FFC4 9000_H	MGDGRPROT0	GRG protection setting register 0	R/W	07FF $FFF0_H$	8/16/32
FFC4 9004_H	MGDGRBAD0	GRG compare base address register 0	R/W	0000 0000_H	8/16/32
FFC4 9008_H	MGDGRADV0	GRG valid compare address register 0	R/W	0000 0000_H	8/16/32
FFC4 9010_H	MGDGRPROT1	GRG protection setting register 1	R/W	07FF $FFF0_H$	8/16/32
FFC4 9014_H	MGDGRBAD1	GRG compare base address register 1	R/W	0000 0000_H	8/16/32
FFC4 9018_H	MGDGRADV1	GRG valid compare address register 1	R/W	0000 0000_H	8/16/32
FFC4 9020_H	MGDGRPROT2	GRG protection setting register 2	R/W	07FF $FFF0_H$	8/16/32
FFC4 9024_H	MGDGRBAD2	GRG compare base address register 2	R/W	0000 0000_H	8/16/32
FFC4 9028_H	MGDGRADV2	GRG valid compare address register 2	R/W	0000 0000_H	8/16/32
FFC4 9030_H	MGDGRPROT3	GRG protection setting register 3	R/W	07FF $FFF0_H$	8/16/32
FFC4 9034_H	MGDGRBAD3	GRG compare base address register 3	R/W	0000 0000_H	8/16/32
FFC4 9038_H	MGDGRADV3	GRG valid compare address register 3	R/W	0000 0000_H	8/16/32
FFC4 9100_H	MGDGRSCTL_VCI	GRG control register (VCI)	R/W	0000 0000_H	8/16/32
FFC4 9104_H	MGDGRSSTAT_VCI	GRG error status register (VCI)	R	0000 0000_H	8/16/32
FFC4 9108_H	MGDGRSAD_VCI	GRG error address register (VCI)	R	0000 0000_H	8/16/32
FFC4 $910C_H$	MGDGRSTYPE_VCI	GRG error access type register (VCI)	R	0000 0000_H	8/16/32
FFC4 9200_H	MGDGRSCTL_PE1	GRG control register (PE1)	R/W	0000 0000_H	8/16/32
FFC4 9204_H	MGDGRSSTAT_PE1	GRG error status register (PE1)	R	0000 0000_H	8/16/32
FFC4 9208_H	MGDGRSAD_PE1	GRG error address register (PE1)	R	0000 0000_H	8/16/32
FFC4 $920C_H$	MGDGRSTYPE_PE1	GRG error access type register (PE1)	R	0000 0000_H	8/16/32

- MGDGRPROTn, MGDGRBADn, and MGDGRADVn set the protection specifications for each channel (n: 0 to 3).
- MGDGRSCTL_*, MGDGRSSTAT_*, MGDGRSAD_*, and MGDGRSTYPE_* indicate error information on each access port: "_VCI" represents access from the system interconnect 1 to the global RAM, and "_PE1" represents access from the CPU1 to the global RAM.

29.4.2.2 Details of Registers

(1) MGDGRPROTn — GRG Protection Setting Register n (n = 0 to 3)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	EN	—	—	—	—	UM	PEID[7:0]								—
Value after reset	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1
R/W	R	R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	SPID[3:0]				DEB	—	—	—	—
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R	R	R	R

Table 29.99 MGDGRPROTn Register Contents

Bit Position	Bit Name	Function
31	—	Reserved This bit is always read as 0. The write value should also be 0.
30	EN	Protection Enable 0: Disables protection. 1: Enables protection. The access permitted by this register is enabled.
29 to 27	—	Reserved These bits are always read as 0. The write value should also be 0.
26	—	Reserved This bit is always read as 1. The write value should also be 1.
25	UM	User Mode Access 0: Enables access in supervisor mode. 1: Enables access in user mode and supervisor mode.
24 to 17	PEID[7:0]	PEID Access The PEID field is a bit list, in which one bit corresponds to one PEID value. Setting multiple bits enables the ID values of the multiple virtual processors simultaneously. For example, setting the PEID field to 0101_B enables access with PEID = 0 and PEID = 2. 0: Disables access with PEID = n. 1: Enables access with PEID = n.
16 to 9	—	Reserved These bits are always read as 1. The write value should also be 1.
8 to 5	SPID[3:0]	SPID Access The SPID field is a bit list, in which one bit corresponds to one SPID value. Setting multiple bits enables the multiple SPID values simultaneously. For example, setting the SPID field to 0101_B enables access with SPID = 0 and SPID = 2. 0: Disables access with SPID = n. 1: Enables access with SPID = n.
4	DEB	Debug Access 0: Disables access from the debug master. 1: Enables access from the debug master. Be sure to set this bit to 1. If the setting is 0, the debugger or RAM monitor tool may not operate correctly.
3 to 0	—	Reserved These bits are always read as 0. The write value should also be 0.

(2) MGDGRBADn — GRG Compare Base Address Register n (n = 0 to 3)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	AD[20:16]				
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	AD[15:9]							—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R

Table 29.100 MGDGRBADn Register Contents

Bit Position	Bit Name	Function
31 to 21	—	Reserved These bits are always read as 0. The write value should also be 0.
20 to 9	AD[20:9]	Compare base address
8 to 0	—	Reserved These bits are always read as 0. The write value should also be 0.

(3) MGDGRADVn — GRG Valid Compare Address Register n (n = 0 to 3)

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	ADV[20:16]				
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ADV[15:9]							—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R

Table 29.101 MGDGRADVn Register Contents

Bit Position	Bit Name	Function
31 to 21	—	Reserved These bits are always read as 0. The write value should also be 0.
20 to 9	ADV[20:9]	Valid Compare Address Address comparison proceeds for addresses for which the corresponding bit in MGDGRADVn[20:9] is 1. When all MGDGRADVn[20:9] bits are set to 1, 512 bytes (the minimum unit) are targeted for protection based on the addresses specified by MGDGRBADn. When all MGDGRADVn[20:9] bits are set to 0, the scope of protection is the whole of the global RAM.
8 to 0	—	Reserved These bits are always read as 0. The write value should also be 0.

Setting example: When the setting of MGDGRBADn[20:9] is 800_H and that of MGDGRADVn[20:9] is $FF7_H$, for example, global RAM guard protection area n is FEF0 0000_H to FEF0 $01FF_H$ and FEF0 1000_H to FEF0 $11FF_H$.

Concept: When the setting of MGDGRBADn[20:9] is 800_H, the base address becomes FEF0 0000_H and the settable range of bits is enclosed in [ ] below.

1111 1110 111 [1 0000 0000 000] 0 0000 0000

F E F 0 0 0 0 0

When MGDGRADVn[20:9] is $FF7_H$, bits with 0 in that range and the 9 lower-order bits are ignored, so we have

1111 1110 111[1 0000 000X 000] X XXXX XXXX

which means that the 512-byte ranges (a total of 1 Kbyte) from

F E F 0 0 0 0 0 to

F E F 0 0 1 F F,

and

F E F 0 1 0 0 0 to

F E F 0 1 1 F F

will be protected.

(4) MGDGRSCTL_VCI/_PE1 — GRG Control Register n

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	ERRCLO	ERRCLE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W

Table 29.102 MGDGRSCTL_VCI/PE1 Register Contents

Bit Position	Bit Name	Function
31 to 2	—	Reserved These bits are always read as 0. The write value should also be 0.
1	ERRCLO	Clears the error entry overflow flag 0: No operation. 1: Clears the overflow flag.
0	ERRCLE	Clears the error detection flag 0: No operation. 1: Clears the error detection flag. Note that this bit must be set simultaneously with the ERRCLO bit as shown in the table below.

Table 29.103 Combination of ERRCLO and ERRCLE

ERRCLO	ERRCLE	Function
0	0	Clears neither of the bits.
0	1	Clears neither of the bits (setting ignored).
1	0	Clears the OVF bit.
1	1	Clears both of the bits.

(5) MGDGRSSTAT_VCI/_PE1 — GRG Error Status Register

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	OVF	ERR
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 29.104 MGDGRSSTAT_VCI/PE1 Register Contents

Bit Position	Bit Name	Function
31 to 2	—	Reserved These bits are always read as 0. The write value should also be 0.
1	OVF	Error Entry Overflow Flag 0: No overflow. 1: Overflow occurred. Since the GRG only has 1 error entry stage, a further guard violation occurring after the error detection flag was set due to an earlier guard violation leads to an error entry overflow and this flag is set. The ECM is also notified when the overflow occurs. Guard violation error information is not captured in case of an overflow.
0	ERR	Error Detection Flag 0: No error 1: An error occurred.

(6) MGDGRSAD_VCI/_PE1 — GRG Error Address Register

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	GRIFREQAP	—	—	—	—	GRIFA[20:16]				
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	GRIFA[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 29.105 MGDGRSAD_VCI/PE1 Register Contents

Bit Position	Bit Name	Function
31 to 26	—	Reserved These bits are always read as 0. The write value should also be 0.
25	GRIFREQAP	Address Parity Bit upon Error Occurrence
24 to 21	—	Reserved These bits are always read as 0. The write value should also be 0.
20 to 0	GRIFA[20:0]	Address upon Error Occurrence

(7) MGDGRSTYPE_VCI/_PE1 — GRG Error Access Type Register

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	PACKETID[1:0]	
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PEID[2:0]			—	—	—	SPID[1:0]		—	UM	—	TYPE[4:0]				
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 29.106 MGDGRSTYPE_VCI/PE1 Register Contents

Bit Position	Bit Name	Function
31 to 18	—	Reserved These bits are always read as 0. The write value should also be 0.
17 and 16	PACKETID[1:0]	Packet ID upon Error Occurrence (Internal bus and interconnect signals. Reference information.)
15 to 13	PEID[2:0]	PEID upon Error Occurrence
12 to 10	—	Reserved These bits are always read as 0. The write value should also be 0.
9 and 8	SPID[1:0]	SPID upon Error Occurrence
7	—	Reserved This bit is always read as 0. The write value should also be 0.
6	UM	UM upon Error Occurrence
5	—	Reserved This bit is always read as 0. The write value should also be 0.
4 to 0	TYPE[4:0]	Transfer Type upon Error Occurrence (Internal bus and interconnect signals. Reference information.)

29.4.3 PBG

The PBG module is divided into several PBG groups, each of which is provided with a maximum of 16 protection channels. A single PBG channel can designate the access against which a single peripheral circuit should be protected. Each PBG group can hold the information of the access that has been rejected.

The following table lists the peripheral circuits to be protected, the corresponding PBG group names, and the PBG channel numbers.

Table 29.107 List of Peripheral Circuit Module To be Protected

PBG Group	PBG Channel Number	Module to be Protected
PBG0A	0	INTC2
	1	DMA_DTS
PBG0B	4	GRG (control register)
	5	GRG status (VCI2GRAM)
	6	GRG status (PE1)
PBG1A	0	APA (except input selector)
	1	APA (input selector)
PBG1B	0	ATU-IV
PBG2A	0	CSIH0 (group A)*1
	1	CSIH0 (group B)*1
	2	CSIH1 (group A)*1
	3	CSIH1 (group B)*1
	4	CSIH2 (group A)*1
	5	CSIH2 (group B)*1
	6	CSIH3 (group A)*1
	7	CSIH3 (group B)*1
PBG3A	0	ΔΣAD (shared register)
	1	ΔΣAD (CH0 register)
	2	ΔΣAD (CH1 register)
	3	ΔΣAD (CH2 register)
	4	ΔΣAD (CH3 register)
	5	ΔΣAD (CH4 register)
	6	ΔΣAD (CH5 register)
	7	ΔΣAD (CH6 register)
	8	ΔΣAD (CH7 register)
	9	ADC0
	10	ADC1
	11	ADC (IFC)
	12	ADC (ASF)
	13	ECM (M)
	14	ECM (C)
	15	ECM (Common)
PBG3B	0	Port group 0 (group A)*2
	1	Port group 0 (group B)*2

Note 1. The CSIHx registers are divided into the following two groups and controlled separately.
Group A: CSIHnCTL0 to CSIHnCTL2, CSIHnSTR0, and CSIHnSTCR0

Group B: Other than above.

Note 2. The port group registers are divided into the following two groups and controlled separately.
Group A: Registers other than PCR0_m registers (see **Section 2.1.4.1 Table 2.6**)
Group B: PCR0_m
Here, m represents the bit numbers (m = 0 to 14) in each port group.

29.4.3.1 List of Registers

The following table lists the register provided for each PBG channel.

Table 29.108 List of Registers (for PBG Channel)

Register Symbol	Register Name	R/W	Value after Reset	Address	Access Size
FSGDxxDPROTn	PBGxx protection register n	R/W	07FF FFFF$_H$	<base_addr0> + 4*n	8/16/32

The following table lists the registers provided for each PBG group.

Table 29.109 List of Registers (for PBG Group)

Register Symbol	Register Name	R/W	Value after Reset	Address	Access Size
ERRSLVxxCTL	PBGxx error control register	W	0000 0000$_H$	<base_addr1> + 0$_H$	8/16/32
ERRSLVxxSTAT	PBGxx error status register	R	0000 0000$_H$	<base_addr1> + 4$_H$	8/16/32
ERRSLVxxADDR	PBGxx error address register	R	0000 0000$_H$	<base_addr1> + 8$_H$	32
ERRSLVxxTYPE	PBGxx error type register	R	0000 0000$_H$	<base_addr1> + C$_H$	16/32

In the above tables, “xx” and “n” in the register names and symbols represents the PBG group numbers and PBG channel numbers, respectively. The table below shows the base address values <base_addr0> and <base_addr1>, which correspond to each of the PBG group numbers and PBG channel numbers.

Table 29.110 List of Base Address

PBG Group	PBG Group Number xx	PBG Channel Number n	<base_addr0>	<base_addr1>
PBG0A	0A	0, 1	FFC4 C000$_H$	FFC4 C800$_H$
PBG0B	0B	4 to 6	FFC4 C100$_H$	FFC4 C900$_H$
PBG1A	1A	0, 1	FFDC 0000$_H$	FFDC 0200$_H$
PBG1B	1B	0	FFDD D000$_H$	FFDD D200$_H$
PBG2A	2A	0 to 7	FFF9 4000$_H$	FFF9 4200$_H$
PBG3A	3A	0 to 15	FFC4 0000$_H$	FFC4 0200$_H$
PBG3B	3B	0, 1	FFC4 1000$_H$	FFC4 1200$_H$

29.4.3.2 Details of Registers

(1) FSGDxxDPROTn — PBGxx Protection Register n

FSGDxxDPROTn designates the access to be rejected against which the peripheral circuit control registers and RAM should be protected. Any access that is disabled using any of the identifiers is rejected as an illegal access.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	PROTUM	PROTPEID[7:0]								—
Value after reset	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	PROTSPID[3:0]				PROTDEB	PROTRDPDEF	PROTWRPDEF	PROTRD	PROTWR
Value after reset	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 29.111 FSGDxxDPROTn Register Contents (1/2)

Bit Position	Bit Name	Function
31 to 27	—	Reserved These bits are always read as 0. The write value should also be 0.
26	—	Reserved This bit is always read as 1. The write value should also be 1.
25	PROTUM	User Mode Access 0: Enables access in supervisor mode. 1: Enables access in user mode and supervisor mode.
24 to 17	PROTPEID[7:0]	PEID Access The PROTnPEID field is a bit list, in which one bit corresponds to one PEID value. Setting multiple bits enables the ID values of the multiple virtual processors simultaneously. For example, setting the PROTnPEID field to 0101_B enables access with PEID = 0 and PEID = 2. 0: Disables access with PEID = n. 1: Enables access with PEID = n.
16 to 9	—	Reserved These bits are always read as 1. The write value should also be 1.
8 to 5	PROTSPID[3:0]	SPID Access The PROTnSPID field is a bit list, in which one bit corresponds to one SPID value. Setting multiple bits enables the multiple SPID values simultaneously. For example, setting the PROTnSPID field to 0101_B enables access with SPID = 0 and SPID = 2. 0: Disables access with SPID = n. 1: Enables access with SPID = n.
4	PROTDEB	Debug Access 0: Disables access from the debug master. 1: Enables access from the debug master. Be sure to set this bit to 1. If the setting of this bit is 0, the debugger or RAM monitor tool may not operate correctly.
3	PROTRDPDEF	Default Read Protection 0: Enables read access from any master. 1: Only enables read access from the master having permitted access.
2	PROTWRPDEF	Default Write Protection 0: Enables write access from any master. 1: Only enables write access from the master having permitted access

Table 29.111 FSGDxxDPROTn Register Contents (2/2)

Bit Position	Bit Name	Function
1	PROTRD	PROTnRD Read Permission 0: Disables read access from any master. 1: Only enables read access from the master having permitted access.
0	PROTWR	PROTnWR Write Permission 0: Disables write access from any master 1: Only enables write access from the master having permitted access.

(2) ERRSLVxxCTL — PBGxx Error Control Register

ERRSLVxxCTL clears the status in the error status register.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CLRO	CLRE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	W	W

Table 29.112 ERRSLVxxCTL Register Contents

Bit Position	Bit Name	Function
31 to 2	—	Reserved These bits are always read as 0. The write value should also be 0.
1	CLRO	Clears the error entry overflow flag. 0: No operation. 1: Clears the overflow flag.
0	CLRE	Clears the error detection flag. 0: No operation. 1: Clears the error detection flag.

(3) ERRSLVxxSTAT — PBGxx Error Status Register

ERRSLVxxSTAT holds the status of the illegal access rejected with the PBGxx.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	OVF	ERR
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 29.113 ERRSLVxxSTAT Register Contents

Bit Position	Bit Name	Function
31 to 2	—	Reserved These bits are always read as 0. The write value should also be 0.
1	OVF	Error Entry Overflow Flag 0: No overflow. 1: Overflow occurred. Since the PBG only has 1 error entry stage, a further guard violation occurring after the error detection flag was set due to an earlier guard violation leads to an error entry overflow and this flag is set. The ECM is not notified of the overflow.
0	ERR	Error Detection Flag 0: No error. 1: Error occurred.

(4) ERRSLVxxADDR — PBGxx Error Address Register

ERRSLVxxADDR holds the address of the illegal access rejected with the PBGxx.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ADDR[31:16]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ADDR[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 29.114 ERRSLVxxADDR Register Contents

Bit Position	Bit Name	Function
31 to 0	ADDR[31:0]	Address at which an error has occurred ADDR[1:0] are always fixed to 0.

(5) ERRSLVxxTYPE — PBGxx Error Type Register

ERRSLVxxTYPE holds the type of the illegal access rejected with the PBGxx.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PEID[2:0]			—	—	—	SPID[1:0]		—	UM	—	STRB[3:0]				WRITE
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 29.115 ERRSLVxxTYPE Register Contents

Bit Position	Bit Name	Function
31 to 16	—	Reserved These bits are always read as 0. The write value should also be 0.
15 to 13	PEID[2:0]	PEID when an error has occurred
12 to 10	—	Reserved These bits are always read as 0. The write value should also be 0.
9, 8	SPID[1:0]	SPID when an error has occurred
7	—	Reserved This bit is always read as 0. The write value should also be 0.
6	UM	UM when an error has occurred
5	—	Reserved This bit is always read as 0. The write value should also be 0.
4 to 1	STRB[3:0]	Strobe signal when an error has occurred (this is an internal bus and interconnect signal and provides information for reference)
0	WRITE	Write signal when an error has occurred (this is an internal bus and interconnect signal and provides information for reference)

29.5 Multi-Input Signature Generator (MISG)

29.5.1 Overview

This LSI incorporates multi-input signature generators (MISG) for self-diagnosis by the CPUs.

The table below shows the overview of the MISG specifications.

Table 29.116 Specification Overview

Item	Description
Generating polynomials	Two polynomials are available for use in signature generation. • MISR1: $G(x) = x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^{8} + x^{7} + x^{5} + x^{4} + x^{2} + x + 1$ • MISR2: $G(x) = x^{32} + x^{22} + x^{2} + x + 1$ *To support signature generation for 64-bit data, each CPU is connected with an MISG that has two MISRs (MISR1 and MISR2).
Signature generation	Signature generation can be enabled or disabled. • Signature generation in MISR1 is enabled or disabled. • Signature generation in MISR2 is enabled or disabled. The following two conditions can be selected as conditions for signature generation • Writing to the register A signature is generated by writing to the MISR calculation register (MISRCDRL) • Monitoring write access A signature is generated if writing to the address area specified for monitoring occurs when writing by the CPU is being monitored. The address area is specified by the monitoring area base address register and the monitoring area address mask register.
Automatic signature comparison	Two signature generation units are selected for comparison of signatures. Each signature generation unit has a data counter and comparison proceeds when the values of the data counters in the MISGs selected as the target for comparison match. The data counter counts the number of write accesses to the MISRCDRL register or the address area being monitored.
Error notification	When signatures are compared and do not match, the error control module is notified of an error. Enabling or disabling of error notification to the error control module can be selected. No interrupt request is not issued directly to the INTC.

29.5.2 Block Diagram

29.5.2.1 MISG

The figure below is a block diagram of the MISG. The MISG consists of two signature generation units and a signature comparison unit.

Write monitoring mode for signature generation conditions (**Section 29.5.3.1 (2), Write Monitoring Mode**) can only be executed between the corresponding signature generation unit and the CPU. Therefore, each signature generation unit is given a name corresponding to the CPU number(MISG_PE1, MISG_PCU). MISG_PE1 can monitor write access by CPU1. MISG_PCU cannot monitor write access by PCU.

When register write mode is the signature generation condition, there is no correspondence between a CPU and the signature generation unit. Any CPU can generate a signature in any signature generation unit.

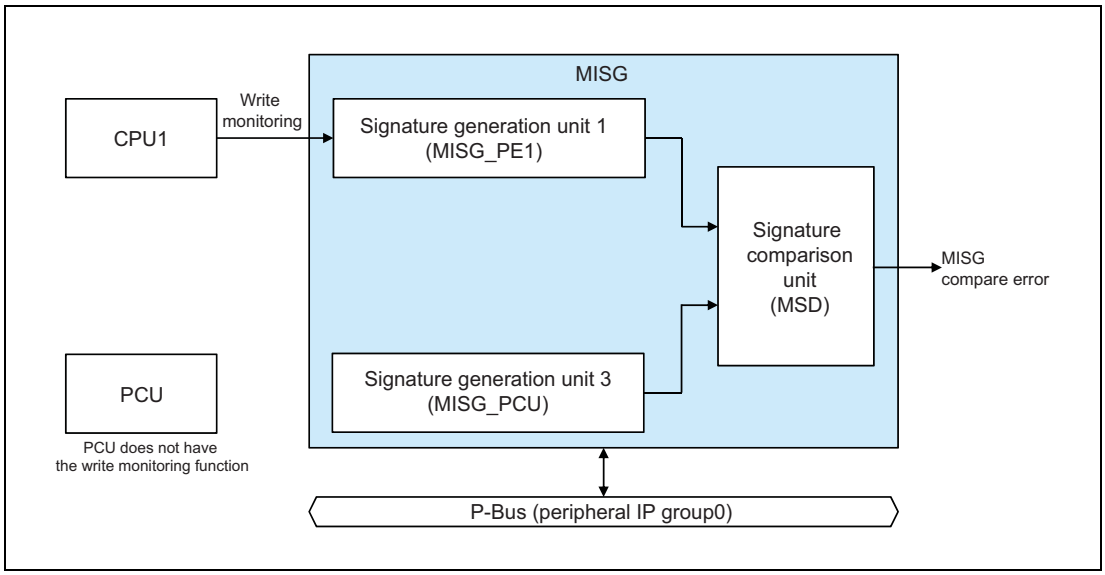


Figure 29.3 MISG Block Diagram (for a Single Core)

29.5.2.2 Signature Generation

The figure below shows the flow of data in signature generation. MISR1 and MISR2 consist of two 32-bit signature generation units (MISR1H and MISR1L, MISR2H and MISR2L).

MISR1H and MISR2H can generate a signature from the 32 higher-order bits of write data of the CPU to be monitored. MISR1L and MISR2L can generate signatures from the 32 lower-order bits of data written by the CPU to be monitored or data written to MISRCDRL.

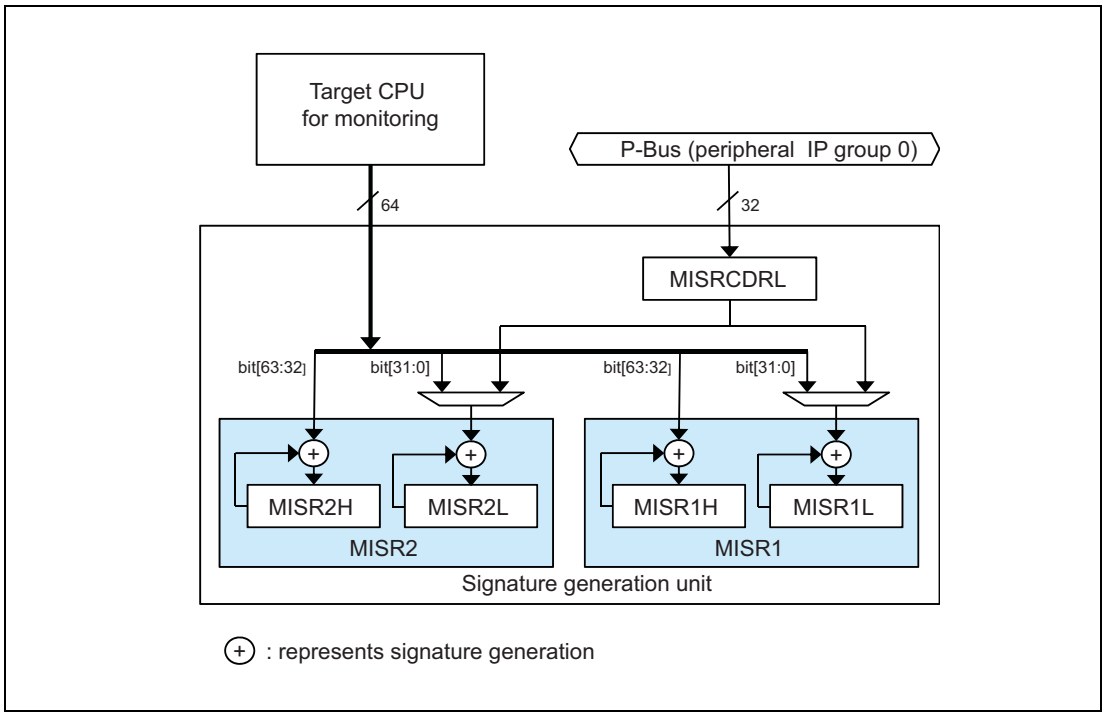


Figure 29.4 Signature Generation Units

Block diagrams of signature generation in MISR1H and MISR1L and the polynomials used for this are shown below.

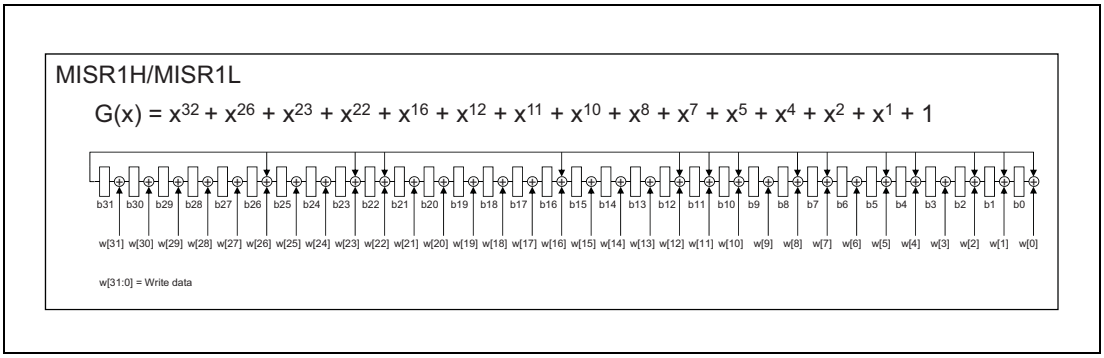


Figure 29.5 Block diagrams of signature generation in MISR1H and MISR1L and the polynomials

Block diagrams of signature generation in MISR2H and MISR2L and the polynomials used for this are shown below.

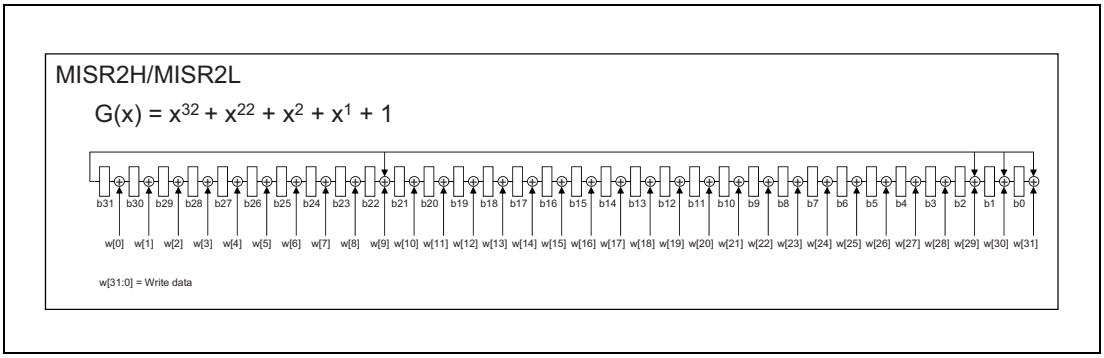


Figure 29.6 Block diagrams of signature generation in MISR2H and MISR2L and the polynomials

29.5.3 Functional Specification

29.5.3.1 Conditions for Signature Generation

The conditions for signature generation in MISR1 and MISR2 can be selected by the setting of the MISR control register (MISRCR).

Signature generation conditions in MISRi (i = 1, 2)

Table 29.117 Signature Generation Conditions of MISRi

MISRCR. MISRiEN	MISRCR. MISEiCND	Signature Generation Conditions
0	—	MISRi does not generate a signature.
1	0	Register write mode Writing to the MISRCDRL register leads to MISRi generating a signature.
1	1	Write monitoring mode MISRi generates a signature after the corresponding CPU writes to an address specified for monitoring.

(1) Register Write Mode

Writing to the MISR calculation register (MISRCDRL) while MISR1 is in register write mode leads to the generation of a 32-bit signature in MISR1 from the value held in the multi-input signature register 1L (MISR1L) and the data written to MISRCDRL, and the result is retained in MISR1L. Similarly, a 32-bit signature is generated in MISR2 from the value held in the multi-input signature register 2L (MISR2L) and the data written to MISRCDRL, and the result is retained in MISR2L. In register write mode, the values of MISR1H and MISR2H are not updated.

Writing to MISRCDRL can proceed in 8-, 16-, or 32-bit units, and bits to which a value is not written are treated as 0. For example, writing to the 16 lower-order bits of MISRCDRL leads to the generation of a signature with the 16 higher-order bits of write data treated as 0. Similarly, writing to the 16 higher-order bits of MISRCDRL leads to the generation of a signature with the 16 lower-order bits of write data treated as 0. However, these cases will not arise if access is by using the IO header file.

MISR1 and MISR2 do not identify the bus master which writes to the MISRCDRL. Writing by any bus master, whether a CPU, DMAC, or debugging master, produces a signature.

(2) Write Monitoring Mode

Writing to the address area specified for monitoring by the CPU while MISR1 is in write monitoring mode leads the generation of two 32-bit signatures in MISR1 from the value held in MISR1H and MISR1L and the data written to the CPU, and the result retained in MISR1H and MISR1L. At this time, MISR1H is for the 32 higher-order bits of write data, while MISR1L is for the 32 lower -order bits of write data. Similarly, writing to the address area specified for monitoring by the CPU while MISR2 is in write monitoring mode leads the generation of two 32-bit signatures in MISR2 from the value held in MISR2H and MISR2L and the data written to the CPU, and the result retained in MISR2H and MISR2L.

The address area specified for monitoring for signature generation is set by the MISR monitoring area base address register (MISRBASEADR) and the MISR monitoring area address mask register (MISRADRMSK). If the CPU write address is within the address area specified for monitoring, a signature is generated.

Write monitoring mode monitors write access by the corresponding CPU in 8-, -16, -32, or -64 units. When writing proceeds in 8-, -16, or -32 units, remaining bits to which a value is not written are treated as 0, and 64-bit data is always input to MISR1 or MISR2. Write data are allocated to the lower-order side regardless of the destination address. For example, when 32 bits are written to an address of the form 8N+4, the 32 bits are allocated to the lower-order side and a signature is generated with the 32 higher-order bits treated as 0.

The CPU write access destinations that can be monitored by MISR1 and MISR2 are as follows:

Local RAM, global RAM, CPU peripheral (local APB),

Peripheral circuit connected to the interconnect (GVCI) or to the P-Bus (global APB)

The following store operations are not subject to monitoring.

(1) Instructions that identify write data in slave responses: BitOp, CAXI, and STC.W

(2) Instructions for saving more than 64 bits on the stack: PREPARE and PUSHSP

Write monitoring mode can only be executed between the specified signature generation unit and the corresponding CPU.

In this product, the following monitoring operations are possible. Writing by the PCU cannot be monitored.

- Monitoring of CPU1 write access by signature generation unit 1

A signature is not generated if write monitoring mode is set for a signature generation unit that does not support write monitoring mode. At this time, reference to the values of MISRBASEADR and MISRADRMSK is not possible from anywhere.

29.5.3.2 Automatic Signature Comparison

Of the signature generation units, two signature generation units are selected for the comparison of signatures by the MISR1CMPEN2 bit or MISR2CMPEN2 bit. Each signature generation unit has a data counter, and the signatures are compared if the values of the data counters generated in two signature generation units selected as targets for comparison match.

If two signatures are to be compared, use MISR1CMPEN2 or MISR2CMPEN2 in the MISRCMPCTL register to enable signature comparison by two signature generation units.

29.5.3.3 Data Counter

The MISR data counter register (MISRDCNT) counts how many times writing to MISRCDRL or the address range specified by BASEADR and ADRMSK proceeds. When the CNTSTA bit = 1 and the CNTTRG bit = 0 in the data counter control register (MISRDCNTCTL), MISRDCNT counts the number of write accesses to MISRCDRL. When the CNTSTA bit = 1 and the CNTTRG bit = 1 in MISRDCNTCTL, MISRDCNT counts the number of write accesses by the corresponding CPU to the address range specified by MISRBASEADR and MISRADRMSK.

MISRDCNT can count the number of times either MISR1 or MISR2 or both MISR1 and MISR2 generate signatures by setting the trigger for counting up by MISRDCNT as the signature generation condition in MISR1 or MISR2. Note, however, that if the signature generation conditions for MISR1

and MISR2 and the trigger for counting up by the data counter are not consistent, the value of MISRDCNT and the signature generation count will not match.

CAUTIONS

1. **When the MISR1EN and MISR2EN bits in the MISRCR register are both 0, counting by MISRDCNT is not incremented even if writing to MISRCDRL and the address range specified by MISRBASEADR and MISRADRMSK proceeded.**
2. **In the signature generation unit that does not support write monitoring mode, counting by MISRDCNT is not incremented when the CPU write access to the monitoring address area is set as the trigger for counting up by the data counter.**

29.5.3.4 Error Notification

When the CMPERREN bit in the error notification control register (MISRERRCTL) is set to 1, the error control module is notified of an error when signatures are compared and do not match. At the same time, the error flag in the compare error status register is set.

An interrupt request for the INTC is not generated.

29.5.4 Register Specifications

29.5.4.1 Register Map

The table below lists the registers of the signature generation units.

The registers with symbols ending in "_PE1" are those of signature generation unit 1 (MISG_PE1).

The registers with symbols ending in "_PCU" are those of signature generation unit 3 (MISG_PCU).

Note that the endings of the register symbols ("_PE1", and "_PCU") are omitted if signature generation units 1 and 3 do not require identification.

MISG_PE1_base = FFC5 1000_H

MISG_PCU_base = FFC5 3000_H

Table 29.118 List of Registers of the Signature Generation Units

Register Symbol	Register Name	R/W	Value after Reset	Address	Access Size		
					8	16	32
MISR1L_PE1	Multi input signature register 1L (PE1)	R/W	0000 0000_H	<MISG_PE1_base> + 000_H			√
MISR1H_PE1	Multi input signature register1H (PE1)	R/W	0000 0000_H	<MISG_PE1_base> + 004_H			√
MISR2L_PE1	Multi input signature register2L (PE1)	R/W	0000 0000_H	<MISG_PE1_base> + 008_H			√
MISR2H_PE1	Multi input signature register2H (PE1)	R/W	0000 0000_H	<MISG_PE1_base> + $00C_H$			√
MISRCDRL_PE1	MISR calculation data register L (PE1)	W	—	<MISG_PE1_base> + 010_H	√	√	√
RFU	RFU (reserved)	—	—	<MISG_PE1_base> + 014_H			
MISRCR_PE1	MISR control register (PE1)	R/W	00_H	<MISG_PE1_base> + 018_H	√		
MISRBASEADR_PE1	MISR monitoring area base address register (PE1)	R/W	0000 0000_H	<MISG_PE1_base> + $01C_H$	√	√	√
MISRADRMSK_PE1	MISR monitoring area mask address register (PE1)	R/W	0000 0000_H	<MISG_PE1_base> + 020_H	√	√	√
MISRDCNTCTL_PE1	MISR data count control register (PE1)	R/W	00_H	<MISG_PE1_base> + 024_H	√		
MISRDCNT_PE1	MISR data count register (PE1)	R/W	0000_H	<MISG_PE1_base> + 028_H		√	
RFU	RFU (reserved)	—	—	<MISG_PE1_base> + $02A_H$ to <MISG_PE1_base> + FFF_H			
MISR1L_PCU	Multi input signature register 1L (PCU)	R/W	0000 0000_H	<MISG_PCU_base> + 000_H			√
MISR1H_PCU	Multi input signature register 1H (PCU)	R/W	0000 0000_H	<MISG_PCU_base> + 004_H			√
MISR2L_PCU	Multi input signature register 2L (PCU)	R/W	0000 0000_H	<MISG_PCU_base> + 008_H			√
MISR2H_PCU	Multi input signature register 2H (PCU)	R/W	0000 0000_H	<MISG_PCU_base> + $00C_H$			√
MISRCDRL_PCU	MISR calculation data register L (PCU)	W	—	<MISG_PCU_base> + 010_H	√	√	√
RFU	RFU (reserved)	—	—	<MISG_PCU_base> + 014_H			
MISRCR_PCU	MISR control register (PCU)	R/W	00_H	<MISG_PCU_base> + 018_H	√		
MISRBASEADR_PCU	MISR monitoring area base address register (PCU)	R/W	0000 0000_H	<MISG_PCU_base> + $01C_H$	√	√	√
MISRADRMSK_PCU	MISR monitoring area mask address register (PCU)	R/W	0000 0000_H	<MISG_PCU_base> + 020_H	√	√	√
MISRDCNTCTL_PCU	MISR data count control register (PCU)	R/W	00_H	<MISG_PCU_base> + 024_H	√		
MISRDCNT_PCU	MISR data count register (PCU)	R/W	0000_H	<MISG_PCU_base> + 028_H		√	
RFU	RFU (reserved)	—	—	<MISG_PCU_base> + $02A_H$ to <MISG_PCU_base> + FFF_H			

The table below lists the registers of the signature comparison unit (MSD sub-block).

MSD_base = FFC5 0000_H

Table 29.119 List of Registers of the Signature Comparison Units

Register symbol	Register Name	R/W	Value after Reset	Address	Access Size		
					8	16	32
MISRCMPCTL	MISR comparator control register	R/W	0000_H	<MSD_base> + 00_H	√	√	
MISRCMPERSTR	MISR compare error status register	R	00_H	<MSD_base> + 04_H	√		
MISRCMPERRSTC	MISR compare error status clear register	W	00_H	<MSD_base> + 08_H	√		
MISRERRCTL	MISR error notification control register	R/W	00_H	<MSD_base> + $0C_H$	√		

Note 1. In case of access to a register in a unit smaller than 32 bits, the bits that are not specified are ignored when writing and read as 0.

The following describes the control registers of the signature comparison unit.

29.5.4.2 MISRCDRL_PE1/PCU — MISR Calculation Data Register

The MISR calculation data register is a 32-bit write-only register.

When signature generation in register write mode is selected, a signature is generated in MISR1 or MISR2 by writing to this register. Data written to this register is the data input to the multi-input signature register 1L (MISR1L) or the multi-input signature register 2L (MISR2L). For the conditions for signature generation, see **Section 29.5.3.1, Conditions for Signature Generation**.

This register can be written in 8-, 16-, or 32-bit units. When writing proceeds in 8- or 16-bit units, remaining bits to which a value is not written are treated as 0, and 32-bit data is always input to MISR1 or MISR2

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MISRCDRL[31:16]															
Value after reset	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MISRCDRL[15:0]															
Value after reset	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
R/W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W	W

Table 29.120 MISRCDRL_PE1/PCU Register Contents

Bit Position	Bit Name	Functions
31 to 0	MISRCDRL31 to MISRCDRL0	Calculation Data Input data to MISR1 or MISR2. When the MISR1EN bit = 1 and the MISR1CND bit = 0 in the MISRCR register, or the MISR2EN bit = 1 and the MISR2CND bit = 0, a new signature is generated each time the MISRCDR register is written, and the result is retained in MISR1 or MISR2.

Note 1. If a signature is generated by writing to this register, the operation is only based on the 32 lower-order bits in MISR1 or MISR2 (MISR1L or MISR2L) and not on the 32 higher-order bits (MISR1H or MISR2H), the values of which remain the same.

29.5.4.3 MISR1L_PE1/PCU — Multi-Input Signature Register 1L

The multi-input signature register 1L is a 32-bit readable and writable register.

Once the signature generation condition is met, this register generates a new signature each time the condition is met and retains the generated value. For the conditions for signature generation, see **Section 29.5.3.1, Conditions for Signature Generation**.

Signatures are generated by using the following polynomial:

$$G(x) = x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^{8} + x^{7} + x^{5} + x^{4} + x^{2} + x + 1$$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MISR1L[31:16]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MISR1L[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 29.121 MISR1L_PE1/PCU Register Contents

Bit Position	Bit Name	Functions
31 to 0	MISR1L31 to MISR1L0	Multi-Input Signature Register 1L When read, a new signature is always read.

29.5.4.4 MISR1H_PE1/PCU — Multi-Input Signature Register 1H

The multi-input signature register 1H is a 32-bit readable and writable register.

Once the signature generation condition is met, this register generates a new signature each time the condition is met and retains the generated value. For the conditions for signature generation, see **Section 29.5.3.1, Conditions for Signature Generation**.

Signatures are generated by using the following polynomial:

$$G(x) = x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x + 1$$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MISR1H[31:16]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MISR1H[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 29.122 MISR1H_PE1/PCU Register Contents

Bit Position	Bit Name	Function
31 to 0	MISR1H31 to MISR1H0	Multi-Input Signature Register 1H When read, a new signature is always read.

Note 1. In register write mode, this register is not updated even when the signature generation condition is met. This register is also provided in the signature generation unit that does not support write monitoring mode, but does not generate signatures.

29.5.4.5 MISR2L_PE1/PCU — Multi-Input Signature Register 2L

The multi-input signature register 2L is a 32-bit readable and writable register.

Once the signature generation condition is met, this register generates a new signature each time the condition is met and retains the generated value. For the conditions for signature generation, see **Section 29.5.3.1, Conditions for Signature Generation**.

Signatures are generated by using the following polynomial:

$$G(x) = x^{32} + x^{22} + x^2 + x + 1$$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MISR2L[31:16]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MISR2L[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 29.123 MISR2L_PE1/PCU Register Contents

Bit Position	Bit Name	Function
31 to 0	MISR2L31 to MISR2L0	Multi-Input Signature Register 2L When read, a new signature is always read.

29.5.4.6 MISR2H_PE1/PCU — Multi-Input Signature Register 2H

The multi-input signature register 2H is a 32-bit readable and writable register.

Once the signature generation condition is met, this register generates a new signature each time the condition is met and retains the generated value. For the conditions for signature generation, see **Section 29.5.3.1, Conditions for Signature Generation**.

Signatures are generated by using the following polynomial:

$$G(x) = x^{32} + x^{22} + x^2 + x + 1$$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	MISR2H[31:16]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	MISR2H[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 29.124 MISR2H_PE1/PCU Register Contents

Bit Position	Bit Name	Functions
31 to 0	MISR2H31 to MISR2H0	Multi-Input Signature Register 2H When read, a new signature is always read.

Note 1. In register write mode, this register is not updated even when the signature generation condition is met. This register is also provided in the signature generation unit that does not support write monitoring mode, but does not generate signatures.

29.5.4.7 MISRCR_PE1/PCU — MISR Control Register

MISR is an 8-bit readable and writable register.

The MISR1EN and MISR2EN bits are used to enable signature generation in MISR1 and MISR2. When the MISR1EN or MISR2EN bit is 1, MISR1 or MISR2 generates a signature and retains the generated value. When the MISR1EN or MISR2EN bit is 0, MISR1 or MISR2 does not generate signatures and the values of these registers are not updated.

When the MISR1EN or MISR2EN bit is 1, the MISR1CND or MISR2CND bit selects the signal generation condition in MISR1 orMISR2. Setting the MISR1CND or MISR2CND bit to 0 selects signature generation in register write mode by MISR1 or MISR2, so writing to MISRCDRL leads to signature generation. Setting the MISR1CND or MISR2CND bit to 1 selects signature generation in write monitoring mode by MISR1 or MISR2, writing by the CPU to the address range specified by the MISRBASEADR and MISRADRMSK registers leads to signature generation. For the conditions for signature generation, see **Section 29.5.3.1, Conditions for Signature Generation**.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	MISR2CND	MISR1CND	MISR2EN	MISR1EN
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 29.125 MISRCR_PE1/PCU Register Contents

Bit Position	Bit Name	Functions
7 to 4	—	Reserved
3	MISR2CND	MISR2 Signature Write Control This bit sets the signature generation condition in MISR2 when the MISR2EN bit = 1. 0: Register write mode 1: Write monitoring mode
2	MISR1CND	MISR1 Signature Write Control This bit sets the signature generation condition in MISR1 when the MISR1EN bit = 1. 0: Register write mode 1: Write monitoring mode
1	MISR2EN	MISR2 Enable 0: MISR2 does not generate a signature. 1: MISR2 generates a signature and the MISR2H and MISR2L values are updated.
0	MISR1EN	MISR1 Enable 0: MISR1 does not generate a signature. 1: MISR1 generates a signature and the MISR1H and MISR1L values are updated.

29.5.4.8 MISRBASEADR_PE1/PCU — MISR Monitoring Area Base Address Register

MISRBASEADR specifies the CPU write access area to be monitored by MISG when signature generation in write monitoring mode is selected. In combination with the setting of the monitoring area mask address register, this register specifies the address range of the monitoring area. For the conditions for signature generation, see **Section 29.5.3.1, Conditions for Signature Generation**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	BASEADR[31:16]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	BASEADR[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 29.126 MISRBASEADR_PE1/PCU Register Contents

Bit Position	Bit Name	Function
31 to 0	BASEADR31 to BASEADR0	Monitoring Area Base Address Register

For the mechanism to judge access to the monitoring area, see **Section 29.5.4.9, MISRADRMSK_PE1/PCU — MISR Monitor Area Address Mask Register**.

29.5.4.9 MISRADRMSK_PE1/PCU — MISR Monitor Area Address Mask Register

MISRADRMSK specifies the CPU write access area to be monitored by MISG when signature generation in write monitoring mode is selected. In combination with the setting of the monitoring area base address register, this register specifies the address range of the monitoring area. For the conditions for signature generation, see **Section 29.5.3.1, Conditions for Signature Generation**.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ADRMSK[31:16]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ADRMSK[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 29.127 ADRMSK_PE1/PCU Register Contents

Bit Position	Bit Name	Function
31 to 0	ADRMSK31 to ADRMSK0	Monitoring Area Mask Address Register

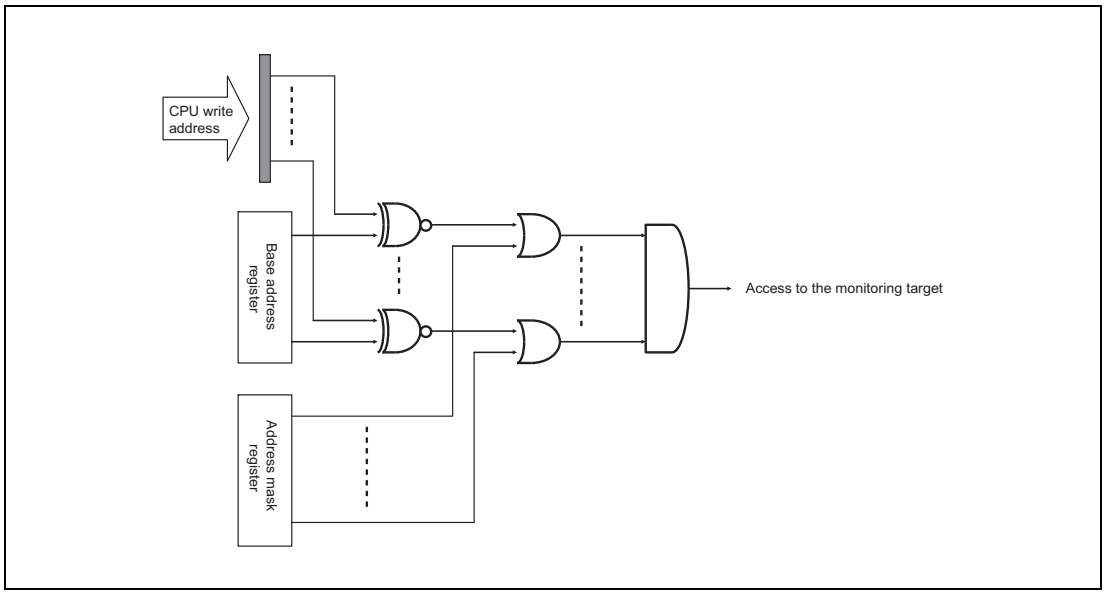


Figure 29.7 How to Detect Access to the Monitoring Area

29.5.4.10 MISRDCNTCTL_PE1/PCU — MISR Data Counter Control Register

This register controls operation of the MSIR data counter register. If the event selected by the CNTTRG bit occurs while the CNTSTA bit is 1, the data counter is incremented. For operation of the data counter, see **Section 29.5.3.3, Data Counter**.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	CNTTRG	CNTSTA
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R/W	R/W

Table 29.128 MISRDCNTCTL_PE1/PCU Register Contents

Bit Position	Bit Name	Functions
7 to 2	—	Reserved
1	CNTTRG	Count-Up Trigger Select This bit selects the trigger for counting up by the data counter. 0: Write access to the MISRCDRL register 1: Write access to the address area specified by the MISRBASEADR and MISRADRMSK registers
0	CNTSTA	Data Counter Start This bit is an enable bit for the data counter. If the event selected by the CNTTRG bit occurs while CNTSTA = 1, the data counter is incremented. 0: The data counter is stopped. 1: The data counter is operating.

29.5.4.11 MISRDCNT_PE1/PCU — MISR Data Counter Register

The data counter is a 16-bit readable and writable register.

If the data counter values of two signature generation units selected for comparison match, automatic comparison of the signatures proceeds.

When the CNTTRG bit in the data counter control register is 0, the data counter is incremented by write access to the MISR calculation data register. When the CNTTRG bit in the data control register is 1, the data counter is incremented by write access by the corresponding CPU to the address area specified by the MISRBASEADR and MISRADRMSK registers. For operation of the data counter, see **Section 29.5.3.3, Data Counter**.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DCNT[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 29.129 MISRDCNT_PE1/PCU Register Contents

Bit Position	Bit Name	Function
15 to 0	DCNT15 to DCNT0	Data Counter Register

The following describes the control registers of the MSD unit (signature comparison unit).

29.5.4.12 MISRCMPCTL — MISR Comparator Control Register

MISRCMPCTL is a 16-bit readable and writable register.

This register controls the comparator that compares signatures generated in the signature generation units. For automatic comparison of signatures, see **Section 29.5.3.2, Automatic Signature Comparison**.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	MISR2CMPEN2	—	—	MISR1CMPEN2	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R/W	R	R	R/W	R	R

Table 29.130 MISRCMPCTL Register Contents

Bit Position	Bit Name	Functions
15 to 6	—	Reserved
5	MISR2CMPEN2	MISR2 Signature Compare Enable 2 This bit controls comparison of signatures retained in MISR2 of MISG_PCU and MISG_PE1. 0: Disables comparison 1: Enables comparison
4, 3	—	Reserved
2	MISR1CMPEN2	MISR1 Signature Compare Enable 2 This bit controls comparison of signatures retained in MISR1 of MISG_PCU and MISG_PE1. 0: Disables comparison 1: Enables comparison
1, 0	—	Reserved

29.5.4.13 MISRCMPERSTR — MISR Compare Error Status Register

The compare error status register is an 8-bit readable register.

If a mismatch occurs in signature comparison enabled by the comparator control register, the corresponding error flag is set.

The error flag is cleared by writing 1 to the corresponding clear bit in the compare error status clear register. The flag is also cleared by a reset.

Bit	7	6	5	4	3	2	1	0
	—	—	MISR2ERR2	—	—	MISR1ERR2	—	—
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 29.131 MISRCMPERSTR Register Contents

Bit Position	Bit Name	Functions
7, 6	—	Reserved
5	MISR2ERR2	MISR2 Signature Compare Error Flag 2 This flag is set when signatures retained in MISR2 of MISG_PE1 and MISG_PCU are compared and do not match. 0: A mismatch has not occurred in signature comparison. 1: A mismatch has occurred in signature comparison.
4, 3	—	Reserved
2	MISR1ERR2	MISR1 Signature Compare Error Flag 2 This flag is set when signatures retained in MISR1 of MISG_PE1 and MISG_PCU are compared and do not match. 0: A mismatch has not occurred in signature comparison. 1: A mismatch has occurred in signature comparison.
1, 0	—	Reserved

29.5.4.14 MISRCMPERRSTC — MISR Compare Error Status Clear Register

The compare error status clear register is an 8-bit write-only register.

When an error flag in the compare error status register is 1, writing 1 to corresponding clear bit clears the error flag. Read the MISR compare error status register and write 1 to the clear bit for the flag being 1.

Bit	7	6	5	4	3	2	1	0
	—	—	MISR2CLR2	—	—	MISR1CLR2	—	—
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	W	R	R	W	R	R

Table 29.132 MISRCMPERRSTC Register Contents

Bit Position	Bit Name	Functions
7, 6	—	Reserved
5	MISR2CLR2	MISR2 Signature Compare Error Clear 2 Writing 1 to this bit clears the MISR2ERR2 bit in the CMPERRST register.
4, 3	—	Reserved
2	MISR1CLR2	MISR1 Signature Compare Error Clear 2 Writing 1 to this bit clears the MISR1ERR2 bit in the CMPERRST register.
1, 0	—	Reserved

29.5.4.15 MISRERRCTL — MISR Error Notification Control Register

The error notification control register is an 8-bit readable and writable register.

This register enables or disables error notification when signatures are compared by the automatic signature comparison and do not mach. For the automatic signature comparison and error notification, see **Section 29.5.3.2, Automatic Signature Comparison**, and **Section 29.5.3.4, Error Notification**

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	CMPERREN
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 29.133 MISRERRCTL Register Contents

Bit Position	Bit Name	Functions
7 to 1	—	Reserved
0	CMPERREN	Compare Error Notification Enable This bit enables or disables error notification when signatures are compared and do not mach. 0: Error notification does not proceed even if a mismatch has occurred in signature comparison. 1: Error notification proceeds if a mismatch has occurred in signature comparison.

29.5.5 Usage

29.5.5.1 Usage Example 1

The self-diagnostic program eases diagnosis. By using the MISG to compress intermediate transitions, the self-diagnostic program makes the saving and comparison of all intermediate results unnecessary. Results of self diagnosis including intermediate transitions can be judged by comparing the results of compression by the MISG with the expected results when the program ends.

This has the effect of reducing requirements for memory capacity and times for processing comparisons (substitution by CRCs is also possible).

Example of settings (when PE1 is running the self-diagnostic program)

The descriptions of registers of an MISG below applies to the registers of signature generation unit 1 (MISG_PE1), which is for PE1.

(1) Initialize the multi-input signature register 1 (MISR1H/MISR1L), multi-input signature register 2 (MISR2H/MISR2L) and data counter register (MISRDCNT).

(2) Set the MISR1CND or MISR2CND bit in the MISR control register (MISRCR) to 1 to select signature generation by MISR1 or MISR2 in write monitoring mode.

(3) Use the MISR monitoring area base address register (MISRBASEADR) and MISR monitoring area address mask register (MISRADRMSK) to set the address area for monitoring.

(4) Set the MISR1EN or MISR2EN bit in the MISR control register to 1 to enable signature generation in MISR1 or MISR2.

(5) Run the self-diagnostic program on PE1.

(6) Upon completion of execution of the self-diagnostic program, MISR1H and MISR1L or MISR2H and MISR2L data is compared with the expected value in flash memory.

29.5.5.2 Usage Example 2

Use multiple processors to run the same processing (including the self-diagnostic program) to confirm the correctness of results. Comparing results from different hardware raises reliability.

Example of settings (when PE1 and PCU are running the same task)

The descriptions of registers of an MISG below applies to the MISG registers of PE1 and PCU.

(1) Initialize the multi-input signature register 1 (MISR1H or MISR1L), multi-input signature register 2 (MISR2H or MISR2L), and data counter register (MISRDCNT).

(2) Set the MISR1CND or MISR2CND bit in the MISR control register (MISRCR) to 0 to select signature generation by MISR1 or MISR2 in register write mode.

(3) Set MISR1CMPEN1 or MISR2CMPEN1 in the MISR comparator control register (MISRCMPCTL) to 1 to enable signature comparison by the comparator.

(4) Set the CMPERREN bit in the MISR error notification control register to 1 to enable error notification to the error control module.

(5) Set the CNTTRG bit in the MISR data counter control register (MISRDCNTCTL) to 0 to set writing to MISRCDRL as the trigger for counting up by the data counter. Set the CNTSTA bit to 1 to enable operation of the data counter.

(6) Set the MISR1EN or MISR2EN bit in the MISR control register to 1 to enable signature generation in MISR1 or MISR2.

(7) Run the self-diagnostic program on all CPUs.

(8) The self-diagnostic program stores intermediate results while the program is running in MISRCDRL of the signature generation units. The signatures in the MISR1 and MISR2 registers of MISG_PE1 and MISG_PCU are compared whenever the values of the data counter registers (MISRDCNT) of MISG_PE1 and MISG_PCU match.

(9) Check the comparison status register to see if there were errors in comparison.

29.6 Clock Monitors

29.6.1 Overview

This product incorporates the clock monitors to monitor the clock operation by detecting the abnormal frequency of the clock to be monitored. The clock monitors provide the following functions.

- Monitors to see if the frequency of the clock to be monitored is within the specified range based on the sampling clock.
- Issues an error notice to the ECM upon detection of the abnormal state of the clock.

Table 29.134 shows the clocks monitored by the clock monitors and the sampling clocks used.

Table 29.134 List of Clocks Monitored by Each Clock Monitor and Sampling Clocks Used

Clock Monitor Channel	Monitored Clock	Sampling Clock
CLMA0	Low-speed peripheral clock (40 MHz SSCG)	Non-modulated low-speed peripheral clock (40 MHz clean)
CLMA1	Non-modulated low-speed peripheral clock (40 MHz clean)	10 MHz (1/2 main OSC)
CLMA2	WDTA counter clock (1/80 main OSC)	Low-speed ROSC clock (ROSC)

29.6.2 List of Registers

29.6.2.1 Clock Monitor Channel Registers

Table 29.135 List of Registers

Address	Symbol	Register Name	R/W	Value after Reset	Access Size
<Base_adr + 00_H>	CLMAnCTL0	CLMAn control register 0	R/W	00_H	8
<Base_adr + 08_H>	CLMAnCMPL	CLMAn compare register L	R/W	0001_H	16
<Base_adr + $0C_H$>	CLMAnCMPH	CLMAn compare register H	R/W	$03FF_H$	16
<Base_adr + 10_H>	CLMAnPCMD	CLMAn protection command register	W	00_H	8
<Base_adr + 14_H>	CLMAnPS	CLMAn protection command status register	R	00_H	8

The base addresses of the registers are shown below.

Table 29.136 List of Base Address Registers

Clock Monitor Channel	<Base_addr>
CLMA0	$FFF8\ 8400_H$
CLMA1	$FFF8\ 8420_H$
CLMA2	$FFF8\ 8440_H$

29.6.2.2 Shared Registers

Table 29.137 List of Shared Registers

Register Symbol	Register Name	R/W	Value after Reset	Address	Access Size	Protection
CLMATEST	CLMA self-test register	R/W	0000 0000_H	FFF8 8204_H	32	PROT1PHCMD *1
CLMATESTS	CLMA self-test status register	R/W	0000 0000_H	FFF8 8208_H	32	

Note 1. For this register, refer to **Section 11, Clock Controller**.

29.6.3 Details of Registers

29.6.3.1 CLMAnCTL0 — CLMAn Control Register 0

CLMAnCTL0 controls the operation of the clock monitors. CLMAnCTL0 is protected by the CLMAnPCMD register.

CLMAnCTL0 can be initialized by either an internal reset or an external reset.

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	CLMAnCLME
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W*1

Note 1. CLMAnCTL0 can be cleared by a reset. Writing 0 is ignored.

Table 29.138 CLMAnCTL0 MISR1CMPEN2 Contents

Bit Position	Bit Name	Function
7 to 1	—	Reserved These bits are always read as 0. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
0	CLMAnCLME	Clock Monitor Operation 0: Disables operation. 1: Enables operation.

29.6.3.2 CLMAnCMPL — CLMAn Compare Register L

CLMAnCMPL sets the lower limit of the normal frequency range used for comparison.

CLMAnCMPL can be initialized by either an internal reset or an external reset.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	CLMAnCMPL[11:0]											
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 29.139 CLMAnCMPL Register Contents

Bit Position	Bit Name	Function
15 to 12	—	Reserved These bits are always read as 0. The write value should also be 0.
11 to 0	CLMAnCMPL [11:0]	Lower Limit of Normal Frequency Range CLMAnCMPL can be written to when CLMAnCTL0.CLMAnCLME is 0. Once CLMAnCTL0.CLMAnCLME is set to 1, writing to CLMAnCMPL is invalid.

29.6.3.3 CLMAnCMPH — CLMAn Compare Register H

CLMAnCMPH sets the upper limit of the normal frequency range used for comparison.

CLMAnCMPH can be initialized by either an internal reset or an external reset.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	CLMAnCMPH[11:0]											
Value after reset	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1
R/W	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 29.140 CLMAnCMPH Register Contents

Bit Position	Bit Name	Function
15 to 12	—	Reserved These bits are always read as 0. The write value should also be 0.
11 to 0	CLMAnCMPH [11:0]	Upper Limit of Normal Frequency Range CLMAnCMPH can be written to when CLMAnCTL0.CLMAnCLME is 0. Once CLMAnCTL0.CLMAnCLME is set to 1, writing to CLMAnCMPH is invalid.

29.6.3.4 CLMAnPCMD — CLMAn Protection Command Register

CLMAnPCMD is a special sequential register for CLMAnCTL0.

CLMAnPCMD can be initialized by either an internal reset or an external reset.

Bit	7	6	5	4	3	2	1	0
	CLMAnPCMD[7:0]							
Value after reset	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W

See **Section 29.6.6.1**, regarding this protection method on details.

29.6.3.5 CLMAnPS — CLMAn Protection Command Status Register

CLMAnPS is a special sequential register for CLMAnCTL0.

CLMAnPS can be initialized by either an internal reset or an external reset

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	CLMAnPRERR
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 29.141 CLMAnPS Register Contents

Bit Position	Bit Name	Function
7 to 1	—	Reserved These bits are always read as 0.
0	CLMAnPRERR	Protection Error Detection 0: A protection error has not been generated. 1: A protection error has been generated.

Operating conditions of the CLMAnPRERR bit:

Setting condition: Access to CLMAnCTL0, which is the target for protection specified by CLMAnPCMD, without following the protection cancelling sequence.

Clearing condition: Writing of A5H to the CLMAnPCMD register (step 1 in the protection cancelling sequence).

29.6.3.6 CLMATEST — CLMA Self-Test Register

CLMATEST is used for self-testing of CLMA2 to CLMA0.

CLMATEST can be protected by the PROT1PHCMD register.

CLMATEST is initialized by an internal reset or an external reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	CLMA2TESTEN	CLMA1TESTEN	CLMA0TESTEN	ERRMSK	MONCLKMSK	RESCLM
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W

Table 29.142 CLMATEST Register Contents

Bit Position	Bit Name	Function
31 to 6	—	Reserved These bits are always read as 0. When writing, follow the procedure of the protection release sequence, write the value after reset, or the bit inverse value of the value after reset to these bits.
5 to 3	CLMA2TESTEN CLMA1TESTEN CLMA0TESTEN	These bits enable or disable self-testing of CLMA2 to CLMA0. 0: Disables self-testing of the corresponding CLMAn. 1: Enables self-testing of the corresponding CLMAn.
2	ERRMSK	Masks an error notification to the ECM when CLMAn detects an error. When the ERRMSK is set for a certain CLMAn, the associated CLMAn does not issue an error notification to the ECM even if it detects an error. ERRMSK setting is valid only for the CLMAn for which CLMAnTESTEN (n = 0 to 2) is set to 1. 0: Does not mask an error notification to the ECM. 1: Masks an error notification to the ECM.
1	MONCLKMSK	Fixes the level of the clock input to the CLMAn that should be monitored, to the low level. MONCLKMSK setting is valid only for the CLMAn for which CLMAnTESTEN (n = 0 to 2) is set to 1. 0: Does not fix the clock input to the CLMAn that should be monitored to the low level. 1: Fixes the clock input to the CLMAn that should be monitored to the low level.
0	RESCLM	Initializes CLMAn forcibly. RESCLM setting is valid only for the CLMAn for which CLMAnTESTEN (n = 0 to 2) is set to 1. 0: Does not initialize CLMAn. 1: Initializes CLMAn.

29.6.3.7 CLMATESTS — CLMA Self-Test Status Register

CLMATESTS indicates the self-testing result of CLMA2 to CLMA0.

CLMATESTS is initialized by an internal reset or an external reset.

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	CLMA2 ERRS	CLMA1 ERRS	CLMA0 ERRS
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 29.143 CLMATESTS Register Contents

Bit Position	Bit Name	Function
31 to 3	—	Reserved These bits are always read as 0.
2 to 0	CLMA2ERRS CLMA1ERRS CLAM0ERRS	These bits indicate whether or not CLMA2 to CLMA0 have detected an error. These bits are not affected by CLMATEST.ERRMSK. 0: The corresponding CLMAn has not yet detected an error. 1: The corresponding CLMAn has detected an error.

29.6.4 Detection of Abnormal Clock Frequency

1. CLMAn counts the number of rising edges of the monitored clock signal within 16 sampling clock cycles, and compares the count with the specified thresholds.
   - The lower threshold is specified using the CLMAnCMPL[11:0] bits in CLMAnCMPL register.
   - The upper threshold is specified using the CLMAnCMPH[11:0] bits in CLMAnCMPH register.
2. When the frequency of the monitored clock is low*1 and the counter value falls below the value set in CLMAnCMPL[11:0] in the CLMAnCMPL register, CLMAn notifies the ECM of the abnormal clock. CLMAn also notifies the ECM of the abnormal clock when the frequency of the monitored clock is high and the counter value exceeds the value set in CLMAnCMPH[11:0] in the CLMAnCMPH register.

Note 1. An abnormality may not be detected when a clock being monitored is completely stopped.

Note that even if the frequency of the monitored clock fluctuates during the sampling period, an error is not notified as long as the number of detected edges falls within the specified range.

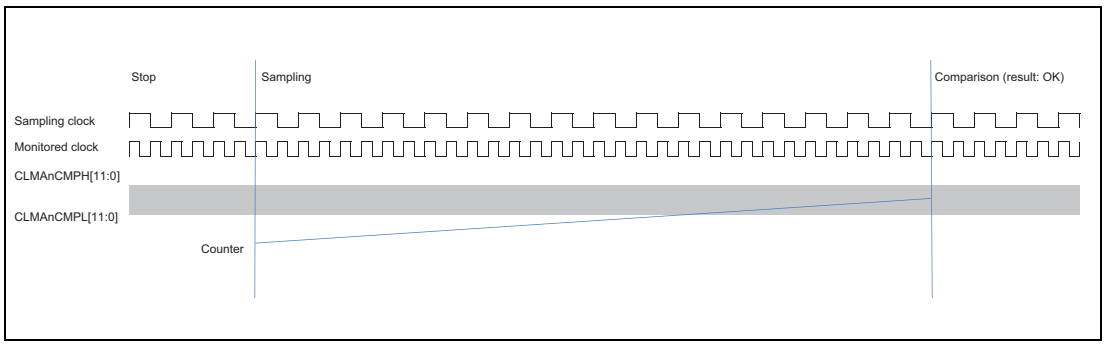


Figure 29.8 Operation When Clock Frequency is Within the Specified Range

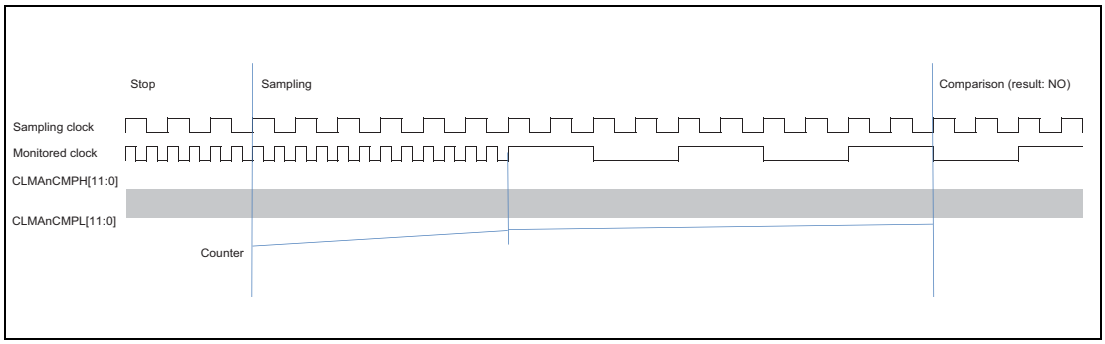


Figure 29.9 Operation When Clock Frequency is Lower than the Specified Range

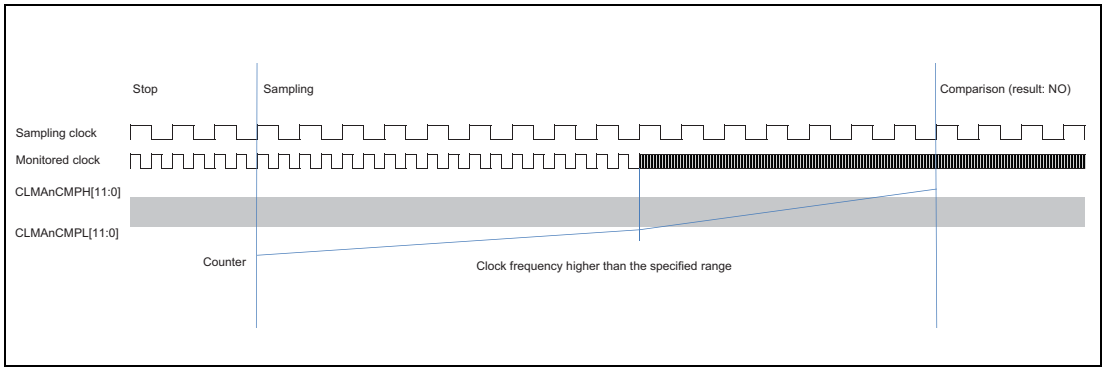


Figure 29.10 Operation When Clock Frequency is Higher than the Specified Range

(1) Calculating the thresholds set by CLMAnCMPL.CLMAnCMPL[11:0] and CLMAnCMPH.CLMAnCMPH[11:0]

The minimum and maximum values of the clock cycles of the monitored clock CLMATMON that are assumed to be valid within 16 cycles of the sampling clock CLMATSMP are set in the comparison registers CLMAnCMPL and CLMAnCMPH.

The expected clock cycle is indicated by “N”.

$$\frac{16}{f_{CLMATSMP}} = \frac{N}{f_{CLMATMON}}$$

$$N = \frac{f_{CLMATMON}}{f_{CLMATSMP}} \times 16$$

The thresholds are calculated considering allowable frequency deviation of CLMATMON and CLMATSMP.

$$\text{Lower threshold} = N_{min} = \frac{f_{CLMATMON(min)}}{f_{CLMATSMP(max)}} \times 16 - 1$$

$$\text{Upper threshold} = N_{max} = \frac{f_{CLMATMON(max)}}{f_{CLMATSMP(min)}} \times 16 + 1$$

NOTE

PLL jitter is covered by “+1” and “-1” as in the formulae.

Example

$f_{CLMATSMP}$ = 240 kHz (±8%) and $f_{CLMATMON}$ = 16 MHz (±5%), the recommendable thresholds will be as follows:

$$N_{min} = 15{,}200 / 259.2 \times 16 - 1$$
$$= 937.27$$
$$CLMAnCMPL = 937 = 03A9_H$$

$$N_{max} = 16{,}800 / 220.8 \times 16 + 1$$
$$= 1218.39$$
$$CLMAnCMPH = 1219 = 04C3_H$$

Minimum Threshold:

The following restrictions must be taken into consideration.

- CLMAnCMPL ≥ 0001_H
- CLMAnCMPH ≥ CLMAnCMPL + 0003_H

(2) Definition of initial values input to the threshold registers

The values of the threshold registers in the initial state and after a reset should permit the maximum frequency deviation of the monitor clock.

- CLMAnCMPL[11:0] = 0001_H
- CLMAnCMPH[11:0] = $03FF_H$

29.6.5 Self-Diagnosis

Self-diagnosis of the clock monitor is available as described below. In the description, the clock monitor to be self-diagnosed is operating.

(1) Set the thresholds for the clock monitor to be subject to self-diagnosis (set CLMAnCMPL and CLMAnCMPH).
At this time, the thresholds must be set such that an error will be produced.

(2) Specify the clock monitor to be self-diagnosed.
Setting the CLMATEST.CLMAnTESEN bit to 1 enables specifying the corresponding clock monitor to be self-diagnosed.

(3) To prevent conveyance of the error signal to the ECM based on self-diagnosis, set CLMATEST.ERRMSK simultaneously with step (2).

(4) Set the CLMAnCTL0.CLMAnCLME bit to 1 to enable operation of the clock monitor.

(5) Wait for the time long enough to allow error occurrence, read the CLMATESTS register to see if an error has been generated in the clock monitor to be self-diagnosed. The time from the start of self-diagnosis to the error occurrence depends on the sampling period. A maximum of two sampling cycles are required.

(6) Clear the error generated by self-diagnosis.
Setting CLMATEST.RESCLM to 1 enables initializing the clock monitor to be self-diagnosed.

(7) Terminate self-diagnosis.
Setting all the bits in CLMATEST to 0 enables terminating self-diagnosis.

Before restarting the clock monitor that has been self-diagnosed, set the registers again as required.

29.6.6 Notes on Register Setting

29.6.6.1 Writing to Protected Registers

Writing to the CLMAnCTL0 register (n = 0 to 2) of each clock monitor is only possible with the protection release sequence described below.

1) Write the fixed value ($A5_H$) to register CLMAnPCMD.

2) Write the new setting to register CLMAnCTL0. Write the value after a reset to the reserved bits.

3) Write the bitwise inverse of the setting value to register CLMAnCTL0. Write the inverse of the value after a reset to the reserved bits.

4) Write the new setting to register CLMAnCTL0. Write the value after a reset to the reserved bits.

A value can only be written to a write-protected register by following the procedure above.

Protection is not released if the procedure is not followed correctly, the setting is not written to the write-protected register, and the CLMAnPS.CLMAnPRERR bit is set to 1. (Although this is not a requirement, you can confirm if the values set in the targeted register was correctly written by checking that the setting of the CLMAnPS.CLMAnPRERR bit is 0 after step 4).)

In case of failure to follow the sequence, repeat the procedure from the beginning.

In case of writing to another register during steps 1) to 4) of the sequence, the protection function operates as follows. Operation is also as described when any interrupt is accepted during the sequence and the interrupt handling includes access to another register.

- When the writing is to another register in the same module*1, writing to the protected register fails and the CLMAnPS.CLMAnPRERR bit is set to 1.
- When writing is to a register in another module, writing to the protected register will successfully proceed to completion, if the remainder of the procedure is followed correctly.

Reading of another register while the sequence is in progress does not lead to failure.

For writing to CLMATEST, see the description of the PROT1PHCMD register.

Note 1. “Another register in the same module” indicates a register under the same module name as that containing the write-protected register. For the module names of the registers, see **Appendix, List of Registers**.

29.6.6.2 Setting CLMAnCMPL/CLMAnCMPH Registers

The CLMAnCMPL/CLMAnCMPH registers should be set so that the following conditions are satisfied. If the clock monitor is otherwise used, operation cannot be guaranteed.

- $1 \leq$ CLMAnCMPL
- CLMAnCMPL + $3 \leq$ CLMAnCMPH (n = 0, 1, or 2)

29.7 BIST

This product incorporates the function to detect failures of the failure detection function itself, which is referred to as BIST. The functions shown in the table below have the dual hardware configuration, and a BIST error notice is issued to the ECM if a failure occurs in any function.

Table 29.144 List of Failure Detection Function

BIST-Applied Function	Remark
Lockstep comparator	Lockstep comparator for CPU1
ECC decoder	
Address parity decoder	

For the ECC decoder for the peripheral RAM and Data Flash, the hardware including the control registers has dual configuration. When using the BIST function, set the same values to the control registers of the master and checker.

The control register is not duplexed for other ECC decoder and address parity checker.

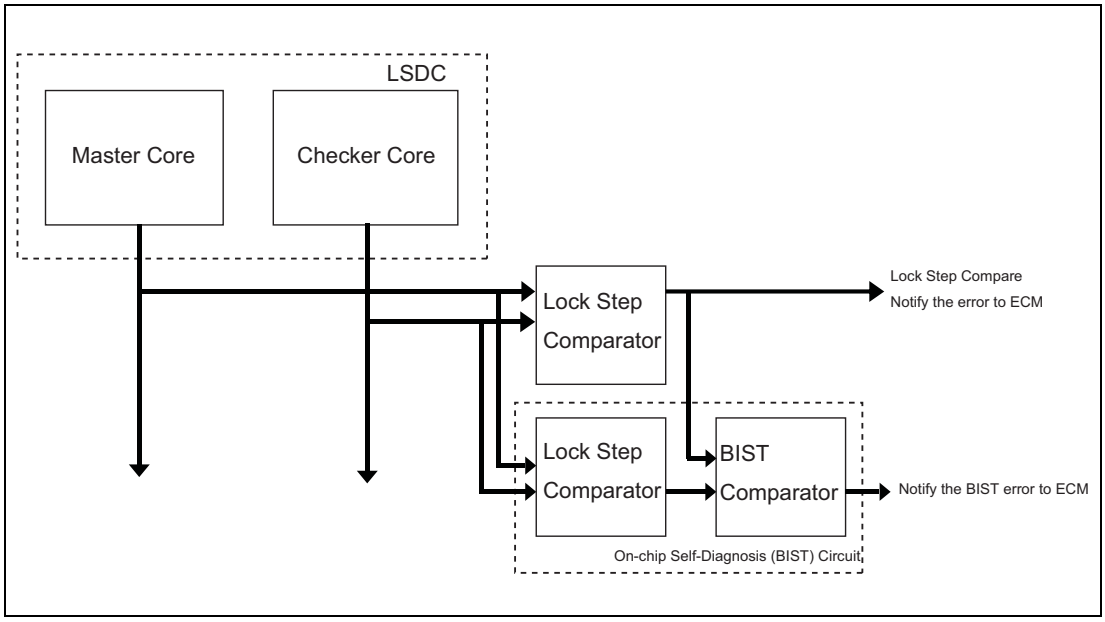


Figure 29.11 Lock Step Configuration

29.8 ECM

The ECM monitors various failure detection states in the LSI chip, and defines the operation to be carried out upon failure detection. For the details of the ECM, refer to **Section 30, Error Control Module (ECM)**.

Section 30 Error Control Module (ECM)

30.1 Overview

30.1.1 Specification Overview

The error control module (ECM) collects error signals coming from different error sources and monitoring circuits. It also outputs error signals (ERROROUT output) from the ERROROUT pin ($\overline{\text{ERROROUT\_M}}$ and $\overline{\text{ERROROUT\_C}}$) and generates interrupts and internal reset signals. **Table 30.1** shows the specification overview of the ECM.

Table 30.1 Specification Overview

Item	Description
Safety processing	The ECM can handle the following processing in response to error signal inputs from individual modules. • Error flag set • Maskable interrupt generation Maskable interrupt generation can be controlled (enabled/disabled) for individual errors. • FE level interrupt generation FE level interrupt generation can be controlled (enabled/disabled) for individual errors. • Internal reset generation Internal reset generation can be controlled (enabled/disabled) for individual errors. • ERROROUT output Pin output mask can be controlled (enabled/disabled) for individual errors. Output can be toggled in response to a timer input or made at a fixed level.
Error status	The ECM incorporates the ECM master/checker error source status register, which can be used to confirm the error status from the error flag. The error flags are only cleared by setting the bit corresponding to the ECM error source status clear trigger register to 1 or an external reset. In case of an internal reset, the error flags are kept and the reset generation source can be confirmed by reading the ECM master/checker error source status register after reset.
Debug, self-diagnosis	• Pseudo errors can be generated for debug and self-diagnosis. The operation during injection of pseudo errors is identical to that for the occurrence of real errors. All configurations for the mask to the ERROROUT output, interrupt, or internal reset apply in the same way. • The ECM incorporates a loop-back function of the ERROROUT output that is used to diagnose the path to the ERROROUT pin. The status of the ERROROUT pin is reflected to the ECM master/checker error source status register and can be confirmed by reading the register.
Timeout function	The ECM incorporates a function that generates an ERROROUT output or internal reset when the count value of the delay timer matches with the delay timer compare register because the delay timer was not stopped during the interrupt processing after being started simultaneously with the occurrence of an interrupt request. The value counted by the delay timer is in cycles of the low-speed peripheral clock.
Register protection	A write-protection with a special sequence is incorporated to protect registers from inadvertent write access.
Others	The ECM is duplexed. The ERROROUT pin is duplexed to the two pins of the mater pin and the checker pin. ERROROUT output of the ECM master and ECM checker are constantly compared. Upon a mismatch, the ECM compare error (error source 26) occurs.

30.1.2 Error Sources and Safety Processing

Table 30.2 shows the error sources and safety processing to the ECM of RH850/E1M-S.

Table 30.2 List of Error Sources and Safety Processing (1/2)

Error Source No.	Module	Error Sources	Error Flag Set	Maskable Interrupt	FE level Interrupt	Internal Reset	ERROR OUT Output	Delay Timer Start
0	WDTA	WDTA error*1	√	√	√	√*2	√	√
1	Reserved		—	—	—	—	—	—
2	Reserved		—	—	—	—	—	—
3	Reserved		—	—	—	—	—	—
4	Lock step	Lock step compare error*3	√	√	√	√	√	√
5	MISG	MISG compare error*3	√	√	√	√	√	√
6	RAM	Local RAM (CPU1, PCU) 2-bit ECC error and local RAM (CPU1) address parity error*3	√	√	√	√	√	√
7		Local RAM (CPU1, PCU) 1-bit ECC error and local RAM (CPU1) address parity error*3	√	√	√	√	√	√
8		Global RAM 2-bit ECC error and address parity error*3	√	√	√	√	√	√
9		Global RAM 1-bit ECC error*3	√	√	√	√	√	√
10		Peripheral RAM (FlexRay, CAN, DFE, DTS, APA) 2-bit ECC error*3	√	√	√	√	√	√
11		Peripheral RAM (FlexRay, CAN, DFE, DTS, APA) 1-bit ECC error*3	√	√	√	√	√	√
12	Code flash	Code flash 2-bit ECC error and address parity error*3	√	√	√	√	√	√
13		Code flash 1-bit ECC error*3	√	√	√	√	√	√
14	Instruction cache	Instruction cache data (CPU1) 2-bit ECC error*3	√	√	√	√	√	√
15		Instruction cache data (CPU1) 1-bit ECC error*3	√	√	√	√	√	√
16		Instruction cache tag (CPU1) 2-bit ECC error*3	√	√	√	√	√	√
17		Instruction cache tag (CPU1) 1-bit ECC error*3	√	√	√	√	√	√
18	Data flash	Data flash 2-bit ECC error*3	√	√	√	√	√	√
19		Data flash 1-bit ECC error*3	√	√	√	√	√	√
20	PE guard function (PEG)	PEG error (CPU1)*4	√	√	√	√	√	√
21	GRG (Global RAM Guard)	GRG error*3	√	√	√	√	√	√
22	PBG	PBG error*3	√	√	√	√	√	√
23	Reserved		—	—	—	—	—	—
24	Reserved		—	—	—	—	—	—
25	Data parity	Data parity error *3	√	√	√	√	√	√
26	ECM	ECM compare error*5	√	√	√	√	√	√

Table 30.2 List of Error Sources and Safety Processing (2/2)

Error Source No.	Module	Error Sources	Error Flag Set	Maskable Interrupt	FE level Interrupt	Internal Reset	ERROR OUT Output	Delay Timer Start
27	Clock monitor	Clock monitor error WDT count clock*3	√	√	√	√	√	√
28		Reserved	—	—	—	—	—	—
29		Reserved	—	—	—	—	—	—
30		Clock monitor error Low-speed peripheral clock*3	√	√	√	√	√	√
31		Clock monitor error Non-modulated low-speed peripheral clock*3	√	√	√	√	√	√
32	Reserved		—	—	—	—	—	—
33	Reserved		—	—	—	—	—	—
34	Reserved		—	—	—	—	—	—
35	Reserved		—	—	—	—	—	—
36	Reserved		—	—	—	—	—	—
37	DSADC	AD parity error*6	√	√	√	√	√	√
	ADC	AD parity error*7	√	√	√	√	√	√
38	Flash	Flash access error*8	√	√	√	√	√	√
39		FACI reset transfer error*8, *11	√	—	—	—	√	—
40	Reserved		—	—	—	—	—	—
41	DMAC	DTS RAM Data ECC SEC-DED*9	√	√	√	√	√	√
42	BIST	Error detection using on-chip self-diagnostic (BIST) circuit*3	√	√	√	√	√	√
43	DMAC	DMAVIOL (DMA violation access notice)*9	√	√	√	√	√	√
44	OSTM	OSTM1 interrupt*10	√	√	√	√	√	√
45	Reserved		—	—	—	—	—	—
46	Reserved		—	—	—	—	—	—
47	Reserved		—	—	—	—	—	—

Note 1. Refer to **Section 19 Window Watchdog Timer (WDTA)** for the details of the errors.
Note 2. The internal reset generation is enabled in the initial state.
Note 3. Refer to **Section 29 Safety** for the details of the errors.
Note 4. Refer to **Section 3 CPU System** for the details of the errors.
Note 5. Refer to **Table 30.1 Specification Overview** for the details of the errors.
Note 6. Refer to **Section 27 Delta-Sigma AD Converter (DS-ADC)** for the details of the errors.
Note 7. Refer to **Section 26 A/D Converter (ADCB)** for the details of the errors.
Note 8. Refer to ***RH850/E1x Flash Memory User's Manual: Hardware Interface*** for the details of the errors.
Note 9. Refer to **Section 7 DMA** for the details of the errors.
Note 10. Refer to **Section 20 OS Timer (OSTM)** for the details of the errors.
Note 11. Operation of the device is not guaranteed if an FACI reset transfer error has occurred. When clearing the error signal output by the ERROROUT pin after release from the reset state, confirm that an FACI reset transfer error has not occurred by checking that an error state is not being indicated again.

Error sources are merged as described in **Table 30.3**.

Table 30.3 Merging of Error Sources

Error Source No	Module	Error Sources	Note
0	WDTA	WDTA error (WDTA0, WDTA1)	Overflows of WDTA0 and WADT1 are merged. The internal reset generation is enabled in the initial state.
6	RAM	Local RAM (CPU1, PCU): 2-bit ECC error, Local RAM (CPU1): Address parity error	2-bit ECC errors (local RAM CPU1, PCU) and address parity errors (excluding PCU) are merged.
7		Local RAM (CPU1, PCU): 1-bit ECC error Local RAM (CPU1): Parity bit error	1-bit ECC errors (local RAM CPU1, PCU) and parity bit errors (excluding PCU) are merged.
8		Global RAM: 2-bit ECC error and address parity error	2-bit ECC errors and address parity errors when global RAM is accessed from each master are merged.
9		Global RAM: 1-bit ECC error	1-bit ECC errors when global RAM is accessed from each master are merged.
10		Peripheral RAM (FlexRay, CAN, DFE, DTS, APA): 2-bit ECC error	2-bit ECC errors for RAM of peripheral circuits are merged.
11		Peripheral RAM (FlexRay, CAN, DFE, DTS, APA): 1-bit ECCerror	1-bit ECC errors for RAM of peripheral circuits are merged.
12	Code flash	Code flash 2-bit ECC error Code flash address parity error	2-bit ECC errors and address parity errors when code flash is accessed from each master are merged.
13		Code flash 1-bit ECC error	1-bit ECC errors when code flash is accessed from each master are merged.
21	GRG (Global RAM Guard)	GRG error	GRG errors from each master in access to the global RAM are merged.
22	PBG	PBG error	Peripheral guard errors for each peripheral circuit are merged.
25	Data parity	Data parity error	Data parity errors for each access path are merged.
37	DSADC	AD parity error	AD parity errors for DSADC0-7 and ADC0-1 are merged.
	ADC	AD parity error	
41	DMA	DTS RAM Data ECC SEC-DED	2-bit ECC errors and 1-bit errors for DTS RAM are merged.
42	BIST	Error detection using on-chip self-diagnostic (BIST) circuit	Errors for each self-diagnostic circuit are merged.
43	DMA	DMAVIOL (DMA violation access notice)	DMA violation accesses for DMAC0, DMAC1, and DTS are merged.

Note: The error sources other than those listed in **Table 30.3** are not merged.

30.1.3 Operations for ERROROUT Output

The $\overline{\text{ERROROUT\_M}}$ pin outputs the error level in the reset state or after release from the reset state. Clear the error output by following the procedure described in **Section 30.3.3**, **CAUTIONS** before use. As the $\overline{\text{ERROROUT\_C}}$ pin is multiplexed with a general-purpose port and other functions, select the $\overline{\text{ERROROUT\_C}}$ function before use. For settings, see **Section 2, Pins**.

ERROROUT output can be set in two operating modes: non-dynamic mode and dynamic mode. ERROROUT output is in synchronization with the occurrence of error source conditions and the error level is output as the pin state regardless of the dynamic mode pulse cycle.

Table 30.4 Operations for ERROROUT Output

Error Status ECMmSSE031 to ECMmSSE000 ECMmSSE115 to ECMmSSE100	Operating Mode ECMSL0 Bit	ERROROUT Output Operating Mode	ERROROUT Output Level	Error Status
0	0	Non-dynamic	High level	Normal
	1	Dynamic	Toggles (along with timer input)*1	Normal
1	0	Non-dynamic	Low level	Error
	1	Dynamic	Low level	Error

Note 1. See **Section 20 OS Timer (OSTM)** for the details.

30.1.3.1 Dynamic Mode Enable

1. Initialize OSTM0.
2. Set the ERROROUT output to normal output by setting the ECMmECT bit (m = M/C) in the ECM master/checker error clear trigger register to 1.
3. Set the ECMSL0 bit in the ECM error pulse configuration register to 1 to specify dynamic mode.
4. Start up OSTM0.

30.1.3.2 Dynamic Mode Disable

1. Set the ERROROUT output to error output by setting the ECMmEST bit (m = M/C) in the ECM master/checker error set trigger register to 1.
2. Stop OSTM0.
3. Clear the ECMSL0 bit in the ECM error pulse configuration register to 0 to specify non-dynamic mode.

30.1.4 Loop-Back Function

The ECM incorporates a loop-back function that is used to check the path to the ERROROUT pin. The output level of the ERROROUT pin can be checked with the ECMmSSE131 bit (m = M/C) in the ECM master/checker error source status register 1.

30.1.5 Pseudo Error Generation

The ECM incorporates a function that can generate pseudo errors for test or debug purposes. The operation of the ECM during injection of pseudo errors is identical to that for the occurrence of real errors. All configurations for error masks, interrupt, internal reset, or delay timer apply in the same way.

30.1.6 Error Status

The error status is indicated by ECM master/checker error source status register 0 and ECM master/checker error source status register 1. The error status is only cleared by setting the bit corresponding to the ECM error source status clear trigger register to 1 or an external reset. In case of an internal reset, the error status is kept and the error of the reset source can be confirmed by reading the ECM master/checker error source status register 0 and ECM master/checker error source status register 1 after release from the reset state.

30.1.7 Write-Protected Registers

Write protected registers are protected from inadvertent write access due to erroneous program execution, etc.

30.1.7.1 Sequence of Writing to the Write-Protected Registers

Writing to the write-protected registers is only possible with the protection unlock sequence described below.

1. Write the fixed value (0000 $00A5_H$) to the ECM protection command register ECMPCMD1 and ECM master/checker protection command register ECMmPCMD0.
2. Write the new setting to the write-protected registers in the ECM and ECMm. Write the value after a reset to the reserved bits.
3. Write the bitwise inverse of the setting value to the same registers as in step 2. Write the inverse of the value after a reset to the reserved bits.
4. Write the new setting to the same registers as in step 2. Write the value after a reset to the reserved bits.

A value can only be written to a write-protected register by following the procedure above.

Protection is not unlocked if the procedure is not followed correctly, the setting is not written to the write-protected register, and the ECMPRERR bit of the ECM protection status register ECMPS is set to 1. (Although this is not a requirement, you can confirm if the values set in the targeted register was correctly written by checking that the setting of the ECMPRERR bit of the ECM protection status register ECMPS is 0 after step 4.)

In case of failure to follow the sequence, repeat the procedure from the beginning.

In case of writing to another register during steps 1 to 4 of the sequence, the protection function operates as follows. Operation is also as described when any interrupt is accepted during the sequence and the interrupt handling includes access to another register.

- When the writing is to another register in the same module*1, writing to the protected register fails and the protection status bit in the protection status register is set to 1.
- When writing is to a register in another module, writing to the protected register will successfully proceed to completion, if the remainder of the procedure is followed correctly.

Reading of another register while the sequence is in progress does not lead to failure.

Note 1. As to ECM, all registers under the module name of ECM, ECMM, or ECMC in **Appendix, List of Registers** are treated as “another register in the same module”.

30.1.8 Timeout Function for Interrupt Processing

The ECM incorporates a function that generates an ERROROUT output or internal reset when the count value of the delay timer incorporated in the ECM matches with the value of the ECM delay timer compare register because the delay timer was not stopped during the interrupt processing after being started simultaneously with the occurrence of an interrupt request. The timer counting is not stopped when a break occurs.

The counting of the delay timer always starts from 0. Configure the duration until an internal reset or ERROROUT output is generated with the settings of the ECM delay timer compare register.

In response to the occurrence of a new error source condition with the setting to start the delay timer while the delay timer is operating, the counter value of the delay timer which is currently operating is not reset. Instead, counting by the timer continues.

30.2 Block Diagram

Figure 30.1 shows the outline of the ECM, and **Figure 30.2** shows connection of the ECM.

The ERROROUT output, internal reset, and ERROROUTZ signals are active low, while the maskable and FE level interrupt signals are active high. Note that the ERROROUTZ signal for PIC1 is not toggled even if the ECM is in dynamic mode.

$\overline{\text{ERROROUT\_M}}$ is ORed with the ERROROUT output of the voltage sensor.

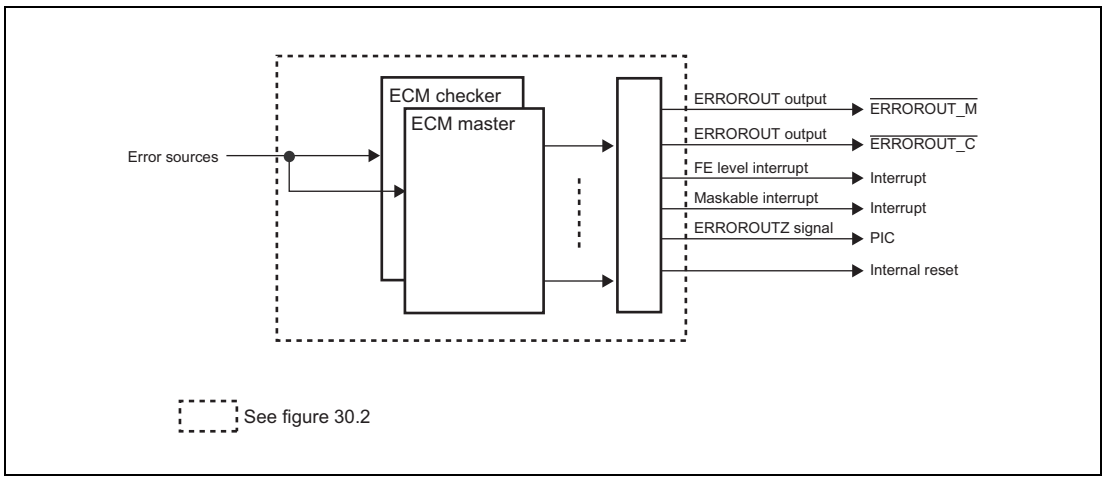


Figure 30.1 Outline of ECM

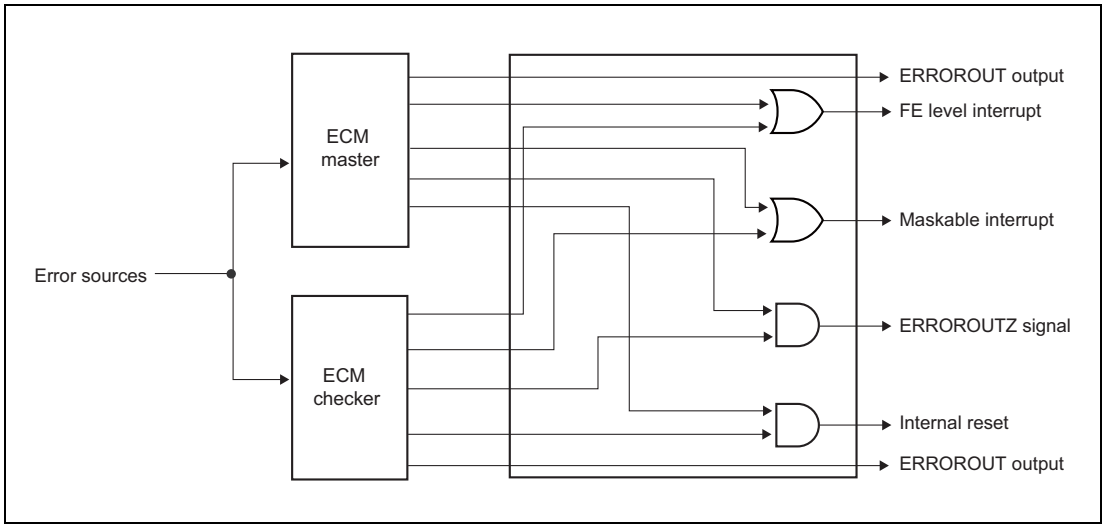


Figure 30.2 Connection of ECM

CAUTION

Pay attention to the difference in output voltage between the $\overline{\text{ERROROUT\_M}}$ and $\overline{\text{ERROROUT\_C}}$ pins because different power supply systems are used for those pins.

$\overline{\text{ERROROUT\_M}}$ pin: VCC (3.3 V)

$\overline{\text{ERROROUT\_C}}$ pin: **EVCC** (5 V) for P4_0 and P4_15; TTLVCC (5 V / 3.3 V) for P13_4

30.3 Register Specification

30.3.1 List of Registers

The ECM registers are divided into three address ranges. These are for the ECM master registers, the ECM checker registers, and the ECM registers common to both master and checker operation (ECM common registers). Writing to the common register area is effective simultaneously for both ECM master and checker. Reading from the common area returns the value for the ECM master. The ECM master registers and ECM checker registers allow separate writing.

The following shows the register map of the ECM master registers.

Table 30.5 List of Registers (ECM master)

<ECMM_base: FFCB 0000$_H$>

Register Symbol	Register Name	R/W	Value after Reset	Protection by Sequence	Address
ECMMESET	ECM master error set trigger register	W	00_H	Protected	<ECMM_base>
ECMMECLR	ECM master error clear trigger register	W	00_H	Protected	<ECMM_base> + 04_H
ECMMESSTR0	ECM master error source status register 0	R	0000 0000_H	Not protected	<ECMM_base> + 08_H
ECMMESSTR1	ECM master error source status register 1	R	0000 0000_H	Not protected	<ECMM_base> + $0C_H$
ECMMPCMD0	ECM master protection command register	W	Undefined	Not protected	<ECMM_base> + 10_H

The following shows the register map of the ECM checker registers.

Table 30.6 List of Registers (ECM checker)

<ECMC_base: FFCB 1000$_H$>

Register Symbol	Register Name	R/W	Value after Reset	Protection by Sequence	Address
ECMCESET	ECM checker error set trigger register	W	00_H	Protected	<ECMC_base>
ECMCECLR	ECM checker error clear trigger register	W	00_H	Protected	<ECMC_base> + 04_H
ECMCESSTR0	ECM checker error source status register 0	R	0000 0000_H	Not protected	<ECMC_base> + 08_H
ECMCESSTR1	ECM checker error source status register 1	R	0000 0000_H	Not protected	<ECMC_base> + $0C_H$
ECMCPCMD0	ECM checker protection command register	W	Undefined	Not protected	<ECMC_base> + 10_H

The following shows the register map of the ECM common registers.

Table 30.7 List of Registers (ECM common)

<ECM_base: FFCB 2000$_H$>

Register Symbol	Register Name	R/W	Value after reset	Protection by Sequence	Address
ECMEPCFG	ECM error pulse configuration register	R/W	00_H	Protected	<ECM_base>
ECMMICFG0	ECM maskable interrupt configuration register 0	R/W	$0000\ 0000_H$	Protected	<ECM_base> + 04_H
ECMMICFG1	ECM maskable interrupt configuration register 1	R/W	$0000\ 0000_H$	Protected	<ECM_base> + 08_H
ECMNMICFG0	ECM FE level interrupt configuration register 0	R/W	$0000\ 0000_H$	Protected	<ECM_base> + $0C_H$
ECMNMICFG1	ECM FE level interrupt configuration register 1	R/W	$0000\ 0000_H$	Protected	<ECM_base> + 10_H
ECMIRCFG0	ECM internal reset configuration register 0	R/W	$0000\ 000F_H$	Protected	<ECM_base> + 14_H
ECMIRCFG1	ECM internal reset configuration register 1	R/W	$0000\ 0000_H$	Protected	<ECM_base> + 18_H
ECMEMK0	ECM error mask register 0	R/W	$0000\ 0000_H$	Protected	<ECM_base> + $1C_H$
ECMEMK1	ECM error mask register 1	R/W	$0000\ 0000_H$	Protected	<ECM_base> + 20_H
ECMESSTC0	ECM error source status clear trigger register 0	W	$0000\ 0000_H$	Protected	<ECM_base> + 24_H
ECMESSTC1	ECM error source status clear trigger register 1	W	$0000\ 0000_H$	Protected	<ECM_base> + 28_H
ECMPCMD1	ECM protection command register	W	Undefined	Not protected	<ECM_base> + $2C_H$
ECMPS	ECM protection status register	R	00_H	Not protected	<ECM_base> + 30_H
ECMPE0	ECM pseudo error trigger register 0	W	$0000\ 0000_H$	Protected	<ECM_base> + 34_H
ECMPE1	ECM pseudo error trigger register 1	W	$0000\ 0000_H$	Protected	<ECM_base> + 38_H
ECMDTMCTL	ECM delay timer control register	W	00_H	Protected	<ECM_base> + $3C_H$
ECMDTMR	ECM delay timer register	R	0000_H	Not protected	<ECM_base> + 40_H
ECMDTMCMP	ECM delay timer compare register	R/W	0000_H	Protected	<ECM_base> + 44_H
ECMDTMCFG0	ECM delay timer configuration register 0	R/W	$0000\ 0000_H$	Protected	<ECM_base> + 48_H
ECMDTMCFG1	ECM delay timer configuration register 1	R/W	$0000\ 0000_H$	Protected	<ECM_base> + $4C_H$
ECMDTMCFG2	ECM delay timer configuration register 2	R/W	$0000\ 0000_H$	Protected	<ECM_base> + 50_H
ECMDTMCFG3	ECM delay timer configuration register 3	R/W	$0000\ 0000_H$	Protected	<ECM_base> + 54_H

30.3.2 ECMmESET (m = M/C) — ECM Master/Checker Error Set Trigger Register

The ECM master/checker error set trigger register is for selecting output of the error signal from the ERROROUT pin. When an ECMmEST bit is set to 1, the ERROROUT pin immediately outputs the error signal. The output cannot be masked. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to **Section 30.1.7, Write-Protected Registers**, for the details of the protection unlock sequence. This register is always read as 00_H.

Access: This register can be written in 8-bit units.

Address: <ECMM_base>
<ECMC_base>

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ECMmESET
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 30.8 ECMmESET Register Contents

Bit Position	Bit Name	Function
7 to 1	—	Reserved When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
0	ECMmESET	Error Set Trigger 0: Writing 0 is invalid 1: Set the output level from the ERROROUT pin to the error output.

CAUTIONS

Setting the ERROROUT output from the ERROROUT pin via the ECMmESET will set the ECMmSSE026 bit of the ECMmESSTR0 register (ECM compare error). Therefore, the ECMmESET or ECMmECLR register has to be set following the sequence below.

1. Set the ECMEMK026 bit of the ECMEMK0 register to “masked”.
2. Prevent the generation of interrupts by setting the ECMMIE026 bit of the ECMMICFG0 register to “prohibited” and the ECMNMIE026 bit of the ECMNMICFG0 register to “prohibited”.
3. Prevent generation of an internal reset by setting the ECMIRE026 bit of the ECMIRCFG0 register to “prohibited”.
4. Set the ERROROUT output bit by the ECMmESET register.
5. Clear error flags by setting the ECMCLSSE026 bit of the ECMESSTC0 register.
6. Make the following settings in accord with the condition of usage for the ECM compare error.
   If an ERROROUT is being output from the ERROROUT pin, set the ECMEMK026 bit of the ECMEMK0 register to “not masked”.
   If an interrupt is being enabled, set the ECMMIE026 bit of the ECMMICFG0 register to “enabled” or the ECMNMIE026 bit of the ECMNMICFG0 register to “enabled”.
   If an internal reset is being enabled, set the ECMIRE026 bit of the ECMIRCFG0 register to “enabled”.

30.3.3 ECMmECLR (m = M/C) — ECM Master/Checker Error Clear Trigger Register

The ECM master/checker error clear trigger register is for setting the error signal from the ERROROUT pin to the normal output. When an ECMmECT bit is set to 1, the ERROROUT pin outputs the normal signal level to indicate that an error has not been detected as long as there are no other sources that set the ERROROUT pin to the normal signal level, which indicates that an error has been detected. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to **Section 30.1.7, Write-Protected Registers**, for the details of the protection unlock sequence. This register is always read as 00_H.

Access: This register can be written in 8-bit units.

Address: <ECMM_base> + 04_H
<ECMC_base> + 04_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ECMmECT
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	W

Table 30.9 ECMmECLR Register Contents

Bit Position	Bit Name	Function
7 to 1	—	Reserved When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
0	ECMmECT	Error Clear Trigger 0: Writing 0 is invalid 1: Set the output level from the ERROROUT pin to the normal output.

CAUTIONS

Clearing of the ERROROUT pin is only possible if all errors, not masked by ECMEMK0/1, are cleared beforehand.

Clearing the ERROROUT output by the ECMmECLR register will set the ECMmSSE026 bit (ECM compare error) in the ECMmESSTR0 register. Therefore, the ECMmECLR register has to be set by following the sequence below.

1. Set the ECMEMK026 (ECM compare error) bit of the ECMEMK0 register to “masked”.
2. Disable interrupts by setting the ECMMIE026 bit of the ECMMICFG0 register to “disabled” and the ECMNMIE026 bit of the ECMNMICFG0 register to “disabled”.
3. Disable an internal reset by setting the ECMIRE026 bit of the ECMIRCFG0 register to “disabled”.
4. Clear error flags by setting the ECMmECLR (m = M/C) register.
5. Clear error flags by setting the ECMCLSSE026 bit of the ECMESSTC0 register.
6. Make the following settings in accordance with the condition of usage for the ECM compare error.
   - If ERROROUT will be output from the ERROROUT pin, set the ECMEMK026 bit in the ECMEMK0 register to "not masked".
   - If interrupts are to be enabled, set the ECMMIE026 bit in the ECMMICFG0 register to “enabled” or the ECMNMIE026 bit in the ECMNMICFG0 register to “enabled”.
   - If a reset exception is to be enabled, set the ECMIRE026 bit in the ECMIRCFG0 register to “enabled”.

30.3.4 ECMmESSTR0 (m = M/C) — ECM Master/Checker Error Source Status Register 0

The ECM master/checker error source status register represents the status of individual internal error sources, which is irrelevant to the setting of the error mask. The status can be cleared only by setting the bit corresponding to the ECM error source status clear trigger register 0 to 1 or by an external reset. When an internal reset will occur, the values of the flags of error source 0 (ECMmSSE000), error resource 4(ECMmSSE004), and error resource 12 (ECMmSSE012) will be retained.

Access: This register can be read in 32-bit units.

Address: <ECMM_base> + 08_H
<ECMC_base> + 08_H

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ECMm SSE031	ECMm SSE030	—	—	ECMm SSE027	ECMm SSE026	ECMm SSE025	—	—	ECMm SSE022	ECMm SSE021	ECMm SSE020	ECMm SSE019	ECMm SSE018	ECMm SSE017	ECMm SSE016
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ECMm SSE015	ECMm SSE014	ECMm SSE016	ECMm SSE012	ECMm SSE011	ECMm SSE010	ECMm SSE009	ECMm SSE008	ECMm SSE007	ECMm SSE006	ECMm SSE005	ECMm SSE004	—	—	—	ECMm SSE000
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 30.10 ECMmESSTR0 Register Contents

Bit Position	Bit Name	Function
31, 30	ECMmSSE031, ECMmSSE030	Error Source Status ECMmSSE031 and ECMmSSE030 correspond to error sources 31 and 30. 0: Error not occurred 1: Error occurred
29, 28	—	Reserved When read, the value after reset is returned.
27 to 25	ECMmSSE027 to ECMmSSE025	Error Source Status ECMmSSE027 to ECMmSSE025 correspond to error sources 27 to 25. 0: Error not occurred 1: Error occurred
24, 23	—	Reserved When read, the value after reset is returned.
22 to 4	ECMmSSE022 to ECMmSSE004	Error Source Status ECMmSSE022 to ECMmSSE004 correspond to error sources 22 to 4. 0: Error not occurred 1: Error occurred
3 to 1	—	Reserved When read, the value after reset is returned.
0	ECMmSSE000	Error Source Status ECMmSSE000 corresponds to error source 0. 0: Error not occurred 1: Error occurred

30.3.5 ECMmESSTR1 (m = M/C) — ECM Master/Checker Error Source Status Register 1

The ECM master/checker error source status register 1 represents the status of individual internal error sources, which is irrelevant to the setting of the error mask. The status can be cleared only by setting the bit corresponding to the ECM error source status clear trigger register 1 to 1 or by an external reset. When an internal reset will occur, The value of flag of the delay timer overflow (ECMmSSE129) will be retained.

Access: This register can be read in 32-bit units.

Address: <ECMM_base> + $0C_H$
<ECMC_base> + $0C_H$

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ECMmSSE131	ECMmSSE130	ECMmSSE129	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	ECMmSSE112	ECMmSSE111	ECMmSSE110	ECMmSSE109	—	ECMmSSE107	ECMmSSE106	ECMmSSE105	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 30.11 ECMmESSTR1 Register Contents

Bit Position	Bit Name	Function
31	ECMmSSE131	Indicates the ERROROUT output loopback status. 0: The ERROROUT_m pin outputs the low level. 1: The ERROROUT_m pin outputs the high level. Note: m = M/C
30	ECMmSSE130	Indicates the ECMmESET write status. 0: No error 1: Error is set by the ECMmEST bit of the ECMmESET register
29	ECMmSSE129	Indicates whether delay timer overflow has occurred. 0: Delay timer overflow not occurred 1: Delay timer overflow occurred
28 to 13	—	Reserved When read, the value after reset is returned.
12 to 9	ECMmSSE112 to ECMmSSE109	Error Source Status ECMmSSE109 to ECMmSSE112 correspond to error sources 44 to 41. 0: Error not occurred 1: Error occurred
8	—	Reserved When read, the value after reset is returned.
7 to 5	ECMmSSE107 to ECMmSSE105	Error Source Status ECMmSSE107 to ECMmSSE105 correspond to error sources 39 to 37. 0: Error not occurred 1: Error occurred
4 to 0	—	Reserved When read, the value after reset is returned.

30.3.6 ECMmPCMD0 (m = M/C) — ECM Master/Checker Protection Command Register

This register is used to protect targeted registers against unauthorized write access such as incorrect programming.

Refer to **Section 30.3.1, List of Registers**, for the protected registers by the ECM master/checker protection command register.

Refer to **Section 30.1.7, Write-Protected Registers**, for the details of the protection unlock sequence.

Access: This register can be written in 32-bit units.

Address: <ECMM_base> + 10_H
<ECMC_base> + 10_H

Value after reset: Undefined

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset								Undefined								
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	ECMmREG0[7:0]							
Value after reset								Undefined								
R/W	R	R	R	R	R	R	R	R	W	W	W	W	W	W	W	W

Table 30.12 ECMmPCMD0 Register Contents

Bit Position	Bit Name	Function
31 to 8	—	Reserved When writing to these bits, write the value after reset.
7 to 0	ECMmREG0 [7:0]	Protection command that enables writing to write protected ECMm registers.

30.3.7 ECMEPCFG — ECM Error Pulse Configuration Register

The ECM error pulse configuration register is used to configure the ERROROUT output operation of ERROROUT pins. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to **Section 30.1.7, Write-Protected Registers**, for the details of the protection unlock sequence.

Access: This register can be read/written in 8-bit units.

Address: <ECM_base>

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ECMSL0
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W

Table 30.13 ECMEPCFG Register Contents

Bit Position	Bit Name	Function
7 to 1	—	Reserved When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
0	ECMSL0	ERROROUT Pin Operation Configuration ERROROUT output operation setting for the ERROROUT pin 0: Non-dynamic mode 1: Dynamic mode

30.3.8 ECMMICFG0 — ECM Maskable Interrupt Configuration Register 0

The ECM maskable interrupt configuration register 0 is used to set the generation of the ECM maskable interrupt. The generation of maskable interrupts in response to errors is selectable. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to **Section 30.1.7, Write-Protected Registers**, for the details of the protection unlock sequence.

Access: This register can be read/written in 32-bit units.

Address: <ECM_base> + 04_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ECMMIE031	ECMMIE030	—	—	ECMMIE027	ECMMIE026	ECMMIE025	—	—	ECMMIE022	ECMMIE021	ECMMIE020	ECMMIE019	ECMMIE018	ECMMIE017	ECMMIE016
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ECMMIE015	ECMMIE014	ECMMIE013	ECMMIE012	ECMMIE011	ECMMIE010	ECMMIE009	ECMMIE008	ECMMIE007	ECMMIE006	ECMMIE005	ECMMIE004	—	—	—	ECMMIE000
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W

Table 30.14 ECMMICFG0 Register Contents

Bit Position	Bit Name	Function
31, 30	ECMMIE031, ECMMIE030	ECM Maskable Interrupt Generation Control ECMMIE031 and ECMMIE030 correspond to error sources 31 and 30. 0: Interrupt generation disabled 1: Interrupt generation enabled
29, 28	—	Reserved When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
27 to 25	ECMMIE027 to ECMMIE025	ECM Maskable Interrupt Generation Control ECMMIE027 to ECMMIE025 correspond to error sources 27 to 25. 0: Interrupt generation disabled 1: Interrupt generation enabled
24, 23	—	Reserved When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
22 to 4	ECMMIE022 to ECMMIE004	ECM Maskable Interrupt Generation Control ECMMIE022 to ECMMIE004 correspond to error sources 22 to 4. 0: Interrupt generation disabled 1: Interrupt generation enabled
3 to 1	—	Reserved When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
0	ECMMIE000	ECM Maskable Interrupt Generation Control ECMMIE000 corresponds to error source 0. 0: Interrupt generation disabled 1: Interrupt generation enabled

30.3.9 ECMMICFG1 — ECM Maskable Interrupt Configuration Register 1

The ECM maskable interrupt configuration register 1 is used to set the generation of maskable interrupts. The generation of maskable interrupts in response to errors is selectable. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to **Section 30.1.7, Write-Protected Registers**, for the details of the protection unlock sequence.

Access: This register can be read/written in 32-bit units.

Address: <ECM_base> + 08_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	ECMMIE112	ECMMIE111	ECMMIE110	ECMMIE109	—	—	ECMMIE106	ECMMIE105	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R	R	R	R	R

Table 30.15 ECMMICFG1 Register Contents

Bit Position	Bit Name	Function
31 to 13	—	Reserved When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
12 to 9	ECMMIE112 to ECMMIE109	ECM Maskable Interrupt Generation Control ECMMIE112 to ECMMIE109 correspond to error sources 44 to 41. 0: Interrupt generation disabled 1: Interrupt generation enabled
8, 7	—	Reserved When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
6, 5	ECMMIE106, ECMMIE105	ECM Maskable Interrupt Generation Control ECMMIE106 and ECMMIE105 correspond to error sources 38 and 37. 0: Interrupt generation disabled 1: Interrupt generation enabled
4 to 0	—	Reserved When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.

30.3.10 ECMNMICFG0 — ECM FE level Interrupt Configuration Register 0

The ECM FE level interrupt configuration register 0 is used to set the generation of ECM FE level interrupts. The generation of FE level interrupts in response to errors is selectable. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to **Section 30.1.7, Write-Protected Registers**, for the details of the protection unlock sequence.

Access: This register can be read/written 32-bit units.

Address: <ECM_base> + $0C_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ECMN MIE031	ECMN MIE030	—	—	ECMN MIE027	ECMN MIE026	ECMN MIE025	—	—	ECMN MIE022	ECMN MIE021	ECMN MIE020	ECMN MIE019	ECMN MIE018	ECMN MIE017	ECMN MIE016
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ECMN MIE015	ECMN MIE014	ECMN MIE013	ECMN MIE012	ECMN MIE011	ECMN MIE010	ECMN MIE009	ECMN MIE008	ECMN MIE007	ECMN MIE006	ECMN MIE005	ECMN MIE004	—	—	—	ECMN MIE000
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W

Table 30.16 ECMNMICFG0 Register Contents

Bit Position	Bit Name	Function
31, 30	ECMNMIE031, ECMNMIE030	ECM FE Level Interrupt Generation Control ECMNMIE031 and ECMNMIE030 correspond to error sources 31 and 30. 0: Interrupt generation disabled 1: Interrupt generation enabled
29, 28	—	Reserved When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
27 to 25	ECMNMIE027 to ECMNMIE025	ECM FE Level Interrupt Generation Control ECMNMIE027 to ECMNMIE025 correspond to error sources 27 to 25. 0: Interrupt generation disabled 1: Interrupt generation enabled
24, 23	—	Reserved When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
22 to 4	ECMNMIE022 to ECMNMIE004	ECM FE Level Interrupt Generation Control ECMNMIE022 to ECMNMIE004 correspond to error sources 22 to 4. 0: Interrupt generation disabled 1: Interrupt generation enabled
3 to 1	—	Reserved When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
0	ECMNMIE000	ECM FE Level Interrupt Generation Control ECMNMIE000 corresponds to error source 0. 0: Interrupt generation disabled 1: Interrupt generation enabled

30.3.11 ECMNMICFG1 — ECM FE level Interrupt Configuration Register 1

The ECM FE level interrupt configuration register 1 is used to set the generation of the ECM FE level interrupt. The generation of FE level interrupts in response to errors is selectable. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to **Section 30.1.7, Write-Protected Registers**, for the details of the protection unlock sequence.

Access: This register can be read/written in 32-bit units.

Address: <ECM_base> + 10_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	ECMN MIE112	ECMN MIE111	ECMN MIE110	ECMN MIE109	—	—	ECMN MIE106	ECMN MIE105	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R	R	R	R	R

Table 30.17 ECMNMICFG1 Register Contents

Bit Position	Bit Name	Function
31 to 13	—	Reserved When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
12 to 9	ECMNMIE112 to ECMNMIE109	ECM FE Level Interrupt Generation Control ECMNMIE112 to ECMNMIE109 correspond to error sources 44 to 41. 0: Interrupt generation disabled 1: Interrupt generation enabled
8, 7	—	Reserved When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
6, 5	ECMNMIE106, ECMNMIE105	ECM FE Level Interrupt Generation Control ECMNMIE106 and ECMNMIE105 correspond to error sources 38 and 37. 0: Interrupt generation disabled 1: Interrupt generation enabled
4 to 0	—	Reserved When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.

30.3.12 ECMIRCFG0 — ECM Internal Reset Configuration Register 0

The ECM internal reset configuration register 0 is used to set the generation of internal resets in response to internal errors. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to **Section 30.1.7, Write-Protected Registers**, for the details of the protection unlock sequence.

Access: This register can be read/written in 32-bit units.

Address: <ECM_base> + 14_H

Value after reset: 0000 $000F_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ECMIRE031	ECMIRE030	—	—	ECMIRE027	ECMIRE026	ECMIRE025	—	—	ECMIRE022	ECMIRE021	ECMIRE020	ECMIRE019	ECMIRE018	ECMIRE017	ECMIRE016
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ECMIRE015	ECMIRE014	ECMIRE013	ECMIRE012	ECMIRE011	ECMIRE010	ECMIRE009	ECMIRE008	ECMIRE007	ECMIRE006	ECMIRE005	ECMIRE004	—	—	—	ECMIRE000
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W

Table 30.18 ECMIRCFG0 Register Contents

Bit Position	Bit Name	Function
31, 30	ECMIRE031, ECMIRE030	ECM Internal Reset Generation Control ECMIRE031 and ECMIRE030 correspond to error sources 31 and 30. 0: Internal reset generation disabled 1: Internal reset generation enabled
29, 28	—	Reserved When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
27 to 25	ECMIRE027 to ECMIRE025	ECM Internal Reset Generation Control ECMIRE027 to ECMIRE025 correspond to error sources 27 to 25. 0: Internal reset generation disabled 1: Internal reset generation enabled
24, 23	—	Reserved When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
22 to 4	ECMIRE022 to ECMIRE004	ECM Internal Reset Generation Control ECMIRE022 to ECMIRE004 correspond to error sources 22 to 4. 0: Internal reset generation disabled 1: Internal reset generation enabled
3 to 1	—	Reserved When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
0	ECMIRE000	ECM Internal Reset Generation Control ECMIRE000 corresponds to error source 0. 0: Internal reset generation disabled 1: Internal reset generation enabled

30.3.13 ECMIRCFG1 — ECM Internal Reset Configuration Register 1

The ECM internal reset configuration register 1 is used to set the generation of internal resets in response to internal errors. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to **Section 30.1.7, Write-Protected Registers**, for the details of the protection unlock sequence.

Access: This register can be read/written in 32-bit units.

Address: <ECM_base> + 18_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	ECMIRE129	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	ECMIRE112	ECMIRE111	ECMIRE110	ECMIRE109	—	—	ECMIRE106	ECMIRE105	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R	R	R	R	R

Table 30.19 ECMIRCFG1 Register Contents

Bit Position	Bit Name	Function
31, 30	—	Reserved When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
29	ECMIRE129	ECM Internal Reset Occurrence Control Corresponds to delay timer overflow. 0: Internal reset generation disabled 1: Internal reset generation enabled
28 to 13	—	Reserved When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
12 to 9	ECMIRE112 to ECMIRE109	ECM Internal Reset Control ECMIRE112 to ECMIRE109 correspond to error sources 44 to 41. 0: Internal reset generation disabled 1: Internal reset generation enabled
8, 7	—	Reserved When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
6, 5	ECMIRE106, ECMIRE105	ECM Internal Reset Control ECMIRE106 and ECMIRE105 correspond to error sources 38 and 37. 0: Internal reset generation disabled 1: Internal reset generation enabled
4 to 0	—	Reserved When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.

30.3.14 ECMEMK0 — ECM Error Mask Register 0

The ECM error mask register 0 is used to set masking of the individual error sources of the ERROROUT output. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to **Section 30.1.7, Write-Protected Registers**, for the details of the protection unlock sequence.

Access: This register can be read/written in 32-bit units.

Address: <ECM_base> + $1C_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ECMEMK031	ECMEMK030	—	—	ECMEMK027	ECMEMK026	ECMEMK025	—	—	ECMEMK022	ECMEMK021	ECMEMK020	ECMEMK019	ECMEMK018	ECMEMK017	ECMEMK016
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ECMEMK015	ECMEMK014	ECMEMK013	ECMEMK012	ECMEMK011	ECMEMK010	ECMEMK009	ECMEMK008	ECMEMK007	ECMEMK006	ECMEMK005	ECMEMK004	—	—	—	ECMEMK000
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W

Table 30.20 ECMEMK0 Register Contents

Bit Position	Bit Name	Function
31, 30	ECMEMK031, ECMEMK030	ECM ERROROUT Output Mask Control ECMEMK031 and ECMEMK030 correspond to error sources 31 and 30. 0: ERROROUT output not masked 1: ERROROUT output masked
29, 28	—	Reserved When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
27 to 25	ECMEMK027 to ECMEMK025	ECM ERROROUT Output Mask Control ECMEMK027 to ECMEMK025 correspond to error sources 27 to 25. 0: ERROROUT output not masked 1: ERROROUT output masked
24, 23	—	Reserved When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
22 to 4	ECMEMK022 to ECMEMK004	ECM ERROROUT Output Mask Control ECMEMK022 to ECMEMK004 correspond to error sources 22 to 4. 0: ERROROUT output not masked 1: ERROROUT output masked
3 to 1	—	Reserved When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
0	ECMEMK000	ECM ERROROUT Output Mask Control ECMEMK000 corresponds to error source 0. 0: ERROROUT output not masked 1: ERROROUT output masked

30.3.15 ECMEMK1 — ECM Error Mask Register 1

The ECM error mask register 1 is used to set masking of the individual error sources of the ERROROUT output. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to **Section 30.1.7, Write-Protected Registers**, for the details of the protection unlock sequence.

Access: This register can be read/written in 32-bit units.

Address: <ECM_base> + 20_H

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	ECME MK129	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	ECME MK112	ECME MK111	ECME MK110	ECME MK109	—	ECME MK107	ECME MK106	ECME MK105	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R	R/W	R/W	R/W	R	R	R	R	R

Table 30.21 ECMEMK1 Register Contents

Bit Position	Bit Name	Function
31, 30	—	Reserved When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
29	ECMEMK129	ECMEMK129 corresponds to an overflow of the delay timer. 0: ERROROUT output not masked 1: ERROROUT output masked
28 to 13	—	Reserved When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
12 to 9	ECMEMK112 to ECMEMK109	ECM ERROROUT Output Mask Control ECMEMK112 to ECMEMK109 correspond to error sources 44 to 41. 0: ERROROUT output not masked 1: ERROROUT output masked
8	—	Reserved When read, the value after reset is returned. When writing to this bit, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
7 to 5	ECMEMK107 to ECMEMK105	ECM ERROROUT Output Mask Control ECMEMK107 to ECMEMK105 correspond to error sources 39 to 37. 0: ERROROUT output not masked 1: ERROROUT output masked
4 to 0	—	Reserved When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.

30.3.16 ECMESSTC0 — ECM Error Source Status Clear Trigger Register 0

The ECM error source status clear trigger register 0 is used to clear the individual error source status of the ECM master/checker error source status register 0. Both the error status of the ECM master and the ECM checker are cleared simultaneously. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to **Section 30.1.7, Write-Protected Registers**, for the details of the protection unlock sequence.

Access: This register can be written in 32-bit units.

Address: <ECM_base> + 24_H

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ECMCLSSE031	ECMCLSSE030	—	—	ECMCLSSE027	ECMCLSSE026	ECMCLSSE025	—	—	ECMCLSSE022	ECMCLSSE021	ECMCLSSE020	ECMCLSSE019	ECMCLSSE018	ECMCLSSE017	ECMCLSSE016
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	R	R	W	W	W	R	R	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ECMCLSSE015	ECMCLSSE014	ECMCLSSE013	ECMCLSSE012	ECMCLSSE011	ECMCLSSE010	ECMCLSSE009	ECMCLSSE008	ECMCLSSE007	ECMCLSSE006	ECMCLSSE005	ECMCLSSE004	—	—	—	ECMCLSSE000
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	R	R	R	W

Table 30.22 ECMESSTC0 Register Contents

Bit Position	Bit Name	Function
31, 30	ECMCLSSE031, ECMCLSSE030	ECM Error Status Clear ECMCLSSE031 and ECMCLSSE030 correspond to error sources 31 and 30. 0: Corresponding error status unchanged 1: Corresponding error status cleared
29, 28	—	Reserved When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
27 to 25	ECMCLSSE027 to ECMCLSSE025	ECM Error Status Clear ECMCLSSE027 to ECMCLSSE025 correspond to error sources 27 to 25. 0: Corresponding error status unchanged 1: Corresponding error status cleared
24, 23	—	Reserved When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
22 to 4	ECMCLSSE022 to ECMCLSSE004	ECM Error Status Clear ECMCLSSE022 to ECMCLSSE004 correspond to error sources 22 to 4. 0: Corresponding error status unchanged 1: Corresponding error status cleared
3 to 1	—	Reserved When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
0	ECMCLSSE000	ECM Error Status Clear ECMCLSSE000 corresponds to error source 0. 0: Corresponding error status unchanged 1: Corresponding error status cleared

30.3.17 ECMESSTC1 — ECM Error Source Status Clear Trigger Register 1

The ECM error source status clear trigger register 1 is used to clear the individual error source status of the ECM master/checker error source status register 1. Both the error status of the ECM master and the ECM checker are cleared simultaneously. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to **Section 30.1.7, Write-Protected Registers**, for the details of the protection unlock sequence.

Access: This register can be written in 32-bit units.

Address: <ECM_base> + 28_H

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	ECMCLSSE130	ECMCLSSE129	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	W	W	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	ECMCLSSE112	ECMCLSSE111	ECMCLSSE110	ECMCLSSE109	—	ECMCLSSE107	ECMCLSSE106	ECMCLSSE105	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	W	W	W	W	R	W	W	W	R	R	R	R	R

Table 30.23 ECMESSTC1 Register Contents

Bit Position	Bit Name	Function
31	—	Reserved When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
30, 29	ECMCLSSE130, ECMCLSSE129	ECM Error Status Clear ECMCLSSE130 and ECMCLSSE129 correspond to writing status and delay timer overflow in the ECMmESET register. 0: Corresponding error status unchanged 1: Corresponding error status cleared
28 to 13	—	Reserved When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
12 to 9	ECMCLSSE112 to ECMCLSSE109	ECM Error Status Clear ECMCLSSE112 to ECMCLSSE109 correspond to error source 44 to 41. 0: Corresponding error status unchanged 1: Corresponding error status cleared
8	—	Reserved When writing to this bit, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
7 to 5	ECMCLSSE107 to ECMCLSSE105	ECM Error Status Clear ECMCLSSE107 to ECMCLSSE105 correspond to error source 39 and 37. 0: Corresponding error status unchanged 1: Corresponding error status cleared
4 to 0	—	Reserved When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.

30.3.18 ECMPCMD1 — ECM Protection Command Register

Refer to **Section 30.3.1, List of Registers**, for the protected registers by the ECM protection command register. Refer to **Section 30.1.7, Write-Protected Registers**, for the details of the protection unlock sequence.

Access: This register can be written in 32-bit units.

Address: <ECM_base> + $2C_H$

Value after reset: Undefined

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset								Undefined								
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	ECMREG1[7:0]							
Value after reset								Undefined								
R/W	R	R	R	R	R	R	R	R	W	W	W	W	W	W	W	W

Table 30.24 ECMPCMD1 Register Contents

Bit Position	Bit Name	Function
31 to 18	—	Reserved When writing to these bits, write the value after reset.
7 to 0	ECMREG1[7:0]	Protection command that enables writing to write protected ECM registers.

30.3.19 ECMPS — ECM Protection Status Register

The ECM protection status register is used to verify the write protected register has been written successfully or not. Refer to **Section 30.1.7, Write-Protected Registers**.

Access: This register can be read in 8-bit units.

Address: <ECM_base> + 30_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	ECMPRERR
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R

Table 30.25 ECMPS Register Contents

Bit Position	Bit Name	Function
7 to 1	—	Reserved When read, the value after reset is returned.
0	ECMPRERR	ECM Protection Status Indicates whether writing to a write protected register is failed or not. 0: Writing was successfully completed. 1: Writing failed

30.3.20 ECMPE0 — ECM Pseudo Error Trigger Register 0

The ECM pseudo error trigger register is used to generate a pseudo error for test purposes. The ECM operation in response to the generation of a pseudo error is identical to that in response to a real error source. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to **Section 30.1.7, Write-Protected Registers**, for the details of the protection unlock sequence.

Access: This register can be written in 32-bit units.

Address: <ECM_base> + 34_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ECMPE031	ECMPE030	—	—	ECMPE027	ECMPE026	ECMPE025	—	—	ECMPE022	ECMPE021	ECMPE020	ECMPE019	ECMPE018	ECMPE017	ECMPE016
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	R	R	W	W	W	R	R	W	W	W	W	W	W	W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ECMPE015	ECMPE014	ECMPE013	ECMPE012	ECMPE011	ECMPE010	ECMPE009	ECMPE008	ECMPE007	ECMPE006	ECMPE005	ECMPE004	—	—	—	ECMPE000
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	W	W	W	W	W	W	W	W	W	W	W	W	R	R	R	W

Table 30.26 ECMPE0 Register Contents

Bit Position	Bit Name	Function
31, 30	ECMPE031, ECMPE030	ECM Pseudo Error Trigger ECMPE031, ECMPE030 correspond to error sources 31 to 30. 0: Pseudo error not generated 1: Generates corresponding pseudo error
29, 28	—	Reserved When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
27 to 25	ECMPE027 to ECMPE025	ECM Pseudo Error Trigger ECMPE027 to ECMPE025 correspond to error sources 27 to 25. 0: Pseudo error not generated 1: Generates corresponding pseudo error
24, 23	—	Reserved When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
22 to 4	ECMPE022 to ECMPE004	ECM Pseudo Error Trigger ECMPE022 to ECMPE004 correspond to error sources 22 to 4. 0: Pseudo error not generated 1: Generates corresponding pseudo error
3 to 1	—	Reserved When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
0	ECMPE000	ECM Pseudo Error Trigger ECMPE000 corresponds to error source 0. 0: Pseudo error not generated 1: Generates corresponding pseudo error

30.3.21 ECMPE1 — ECM Pseudo Error Trigger Register 1

The ECM pseudo error trigger register 1 is used to generate a pseudo error for test purposes. The ECM operation in response to the generation of a pseudo error is identical to that in response to a real error source. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to **Section 30.1.7, Write-Protected Registers**, for the details of the protection unlock sequence.

Access: This register can be written in 32-bit units.

Address: <ECM_base> + 38_H

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	ECMPE129	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	W	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	ECMPE112	ECMPE111	ECMPE110	ECMPE109	—	ECMPE107	ECMPE106	ECMPE105	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	W	W	W	W	R	W	W	W	R	R	R	R	R

Table 30.27 ECMPE1 Register Contents

Bit Position	Bit Name	Function
31, 30	—	Reserved When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
29	ECMPE129	ECM Pseudo Error Trigger Corresponds to delay timer overflow. 0: Pseudo error not generated 1: Generates corresponding pseudo error
28 to 13	—	Reserved When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
12 to 9	ECMPE112 to ECMPE109	ECM Pseudo Error Trigger ECMPE112 to ECMPE109 correspond to error sources 44 to 41. 0: Pseudo error not generated 1: Generates corresponding pseudo error
8	—	Reserved When writing to this bit, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
7 to 5	ECMPE107 to ECMPE105	ECM Pseudo Error Trigger ECMPE107 to ECMPE105 correspond to error sources 39 to 37. 0: Pseudo error not generated 1: Generates corresponding pseudo error
4 to 0	—	Reserved When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.

30.3.22 ECMDTMCTL — ECM Delay Timer Control Register

The ECM delay timer control register is used to control the delay timer. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to **Section 30.1.7, Write-Protected Registers**, for the details of the protection unlock sequence.

Access: This register can be written in 8-bit units.

Address: <ECM_base> + $3C_H$

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	ECMSTP	ECMSTA
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	W	W

Table 30.28 ECMDTMCTL Register Contents

Bit Position	Bit Name	Function
7 to 2	—	Reserved When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
1	ECMSTP	Delay Timer Stop Writing 1 to this bit initializes the delay timer counter, causing the delay timer to stop. Simultaneously, the ECMSTA bit is set to 0.
0	ECMSTA	Delay Timer Stop Writing 1 to this bit starts the delay timer when an interrupt is generated. (Writing 0 to this bit stops the delay timer.)

30.3.23 ECMDTMR — ECM Delay Timer Register

The ECM delay timer register represents the counter value of the delay timer.

The delay timer counter is initialized by writing 1 to the ECMSTP bit or writing 0 to the ECMSTA bit of the ECM delay timer control register (ECMDTMCTL).

Access: This register can be read in 16-bit units.

Address: <ECM_base> + 40_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ECMDTMR[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

30.3.24 ECMDTMCMP — ECM Delay Timer Compare Register

A delay timer overflow signal is generated to set the ECMmSSE129n bit when this register matches with the value of the delay timer counter. Writing data to this register has to be conducted while the delay timer is stopped. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to **Section 30.1.7, Write-Protected Registers**, for the details of the protection unlock sequence.

Access: This register can be read/written in 16-bit units.

Address: <ECM_base> + 44_H

Value after reset: 0000_H

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ECMDTMCMP[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

30.3.25 ECMDTMCFG0 — ECM Delay Timer Configuration Register 0

The ECM delay timer configuration register 0 is used to set enable/disable of the delay timer start caused by maskable interrupts in response to errors. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to **Section 30.1.7, Write-Protected Registers**, for the details of the protection unlock sequence.

Access: This register can be read/written in 32-bit units.

Address: <ECM_base> + 48_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ECMTE031	ECMTE030	—	—	ECMTE027	ECMTE026	ECMTE025	—	—	ECMTE022	ECMTE021	ECMTE020	ECMTE019	ECMTE018	ECMTE017	ECMTE016
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ECMTE015	ECMTE014	ECMTE013	ECMTE012	ECMTE011	ECMTE010	ECMTE009	ECMTE008	ECMTE007	ECMTE006	ECMTE005	ECMTE004	—	—	—	ECMTE000
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W

Table 30.29 ECMDTMCFG0 Register Contents (1/2)

Bit Position	Bit Name	Function
31, 30	ECMTE031, ECMTE030	ECM Delay Timer Start Control ECMTE031 and ECMTE030 correspond to maskable interrupts generated by error sources 31 and 30. 0: Delay timer start disabled 1: Delay timer start enabled
29, 28	—	Reserved When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
27 to 25	ECMTE027 to ECMTE025	ECM Delay Timer Start Control ECMTE027 to ECMTE025 correspond to maskable interrupts generated by error sources 27 to 25. 0: Delay timer start disabled 1: Delay timer start enabled
24, 23	—	Reserved When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
22 to 4	ECMTE022 to ECMTE004	ECM Delay Timer Start Control ECMTE022 to ECMTE004 correspond to maskable interrupts generated by error sources 22 to 4. 0: Delay timer start disabled 1: Delay timer start enabled
3 to 1	—	Reserved When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.

Table 30.29 ECMDTMCFG0 Register Contents (2/2)

Bit Position	Bit Name	Function
0	ECMTE000	ECM Delay Timer Start Control ECMTE000 corresponds to the maskable interrupt generated by error source 0. 0: Delay timer start disabled 1: Delay timer start enabled

30.3.26 ECMDTMCFG1 — ECM Delay Timer Configuration Register 1

The ECM delay timer configuration register 1 is used to set enable/disable of the delay timer start caused by maskable interrupts in response to errors. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to **Section 30.1.7, Write-Protected Registers**, for the details of the protection unlock sequence.

Access: This register can be read/written in 32-bit units.

Address: <ECM_base> + $4C_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	ECMTE112	ECMTE111	ECMTE110	ECMTE109	—	—	ECMTE106	ECMTE105	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R	R	R	R	R

Table 30.30 ECMDTMCFG1 Register Contents

Bit Position	Bit Name	Function
31 to 13	—	Reserved When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
12 to 9	ECMTE112 to ECMTE109	ECM Delay Timer Start Control ECMTE112 to ECMTE109 correspond to maskable interrupts generated by error sources 44 to 41. 0: Delay timer start disabled 1: Delay timer start enabled
8, 7	—	Reserved When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
6, 5	ECMTE106, ECMTE105	ECM Delay Timer Start Control ECMTE106 and ECMTE105 correspond to maskable interrupts generated by error sources 38 and 37. 0: Delay timer start disabled 1: Delay timer start enabled
4 to 0	—	Reserved When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.

30.3.27 ECMDTMCFG2 — ECM Delay Timer Configuration Register 2

The ECM delay timer configuration register 2 is used to set enable/disable of the delay timer start caused by FE level interrupts in response to errors. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to **Section 30.1.7, Write-Protected Registers**, for the details of the protection unlock sequence.

Access: This register can be read/written in 32-bit units.

Address: <ECM_base> + 50_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	ECMTE231	ECMTE230	—	—	ECMTE227	ECMTE226	ECMTE225	—	—	ECMTE222	ECMTE221	ECMTE220	ECMTE219	ECMTE218	ECMTE217	ECMTE216
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R	R	R/W	R/W	R/W	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	ECMTE215	ECMTE214	ECMTE213	ECMTE212	ECMTE211	ECMTE210	ECMTE209	ECMTE208	ECMTE207	ECMTE206	ECMTE205	ECMTE204	—	—	—	ECMTE200
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R	R	R	R/W

Table 30.31 ECMDTMCFG2 Register Contents (1/2)

Bit Position	Bit Name	Function
31, 30	ECMTE231, ECMTE230	ECM Delay Timer Start Control ECMTE231 and ECMTE230 correspond to FE level interrupts generated by error sources 31 and 30. 0: Delay timer start disabled 1: Delay timer start enabled
29, 28	—	Reserved When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
27 to 25	ECMTE227 to ECMTE225	ECM Delay Timer Start Control ECMTE227 to ECMTE225 correspond to FE level interrupts generated by error sources 27 to 25. 0: Delay timer start disabled 1: Delay timer start enabled
24, 23	—	Reserved When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
22 to 4	ECMTE222 to ECMTE204	ECM Delay Timer Start Control ECMTE222 to ECMTE204 correspond to FE level interrupts generated by error sources 22 to 4. 0: Delay timer start disabled 1: Delay timer start enabled
3 to 1	—	Reserved When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.

Table 30.31 ECMDTMCFG2 Register Contents (2/2)

Bit Position	Bit Name	Function
0	ECMTE200	ECM Delay Timer Start Control ECMTE200 corresponds to the FE level interrupt generated by error source 0. 0: Delay timer start disabled 1: Delay timer start enabled

30.3.28 ECMDTMCFG3 — ECM Delay Timer Configuration Register 3

The ECM delay timer configuration register 3 is used to set enable/disable of the delay timer start caused by FE level interrupts in response to errors. You have to follow a predetermined protection unlock sequence for writing data to this register. Refer to **Section 30.1.7, Write-Protected Registers**, for the details of the protection unlock sequence.

Access: This register can be read/written in 32-bit units.

Address: <ECM_base> + 54_H

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	ECMTE312	ECMTE311	ECMTE310	ECMTE309	—	—	ECMTE306	ECMTE305	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R/W	R/W	R/W	R/W	R	R	R/W	R/W	R	R	R	R	R

Table 30.32 ECMDTMCFG3 Register Contents

Bit Position	Bit Name	Function
31 to 13	—	Reserved When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
12 to 9	ECMTE312 to ECMTE309	ECM Delay Timer Start Control ECMTE312 to ECMTE309 correspond to FE level interrupts generated by error sources 44 to 41. 0: Delay timer start disabled 1: Delay timer start enabled
8, 7	—	Reserved When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.
6, 5	ECMTE306, ECMTE305	ECM Delay Timer Start Control ECMTE306 and ECMTE305 correspond to FE level interrupts generated by error sources 38 and 37. 0: Delay timer start disabled 1: Delay timer start enabled
4 to 0	—	Reserved When read, the value after reset is returned. When writing to these bits, follow the procedure of the protection unlock sequence to write the value after a reset or the bitwise inverse of the value after a reset.

Section 31 Data CRC (DCRA)

This section explains the data CRC function A (DCRA).

Table 31.1 shows the register base address.

Table 31.1 Register Base Address

Base Address Name	Base Address
<DCRA0_base>	FFF7 0000_H

31.1 Overview

31.1.1 Functional Overview

The data CRC function A can verify or generate data streams protected by CRC with various lengths and different bit widths.

- Generating polynomial: Either 32-Ethernet (32-bit CRC code generation) or 16-CCITT (16-bit CRC code generation) can be selected with a register.
  The generating polynomials are as follows:
  - 32-Ethernet
    $(x^{32} + x^{26} + x^{23} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^8 + x^7 + x^5 + x^4 + x^2 + x + 1)$
  - 16-CCITT $(x^{16} + x^{12} + x^5 + 1)$
- When 32-Ethernet is selected, a 32-bit CRC code is generated to a data block of an arbitrary length in units of 8/16/32 bits as specified with a register.
- When 16-CCITT is selected, a 16-bit CRC code is generated to a data block of an arbitrary length in units of 8/16/32 bits as specified with a register.
- After setting an initial value in the CRC data register (DCRA0COUT), the calculation is carried out by transferring an input data to the CRC input register (DCRA0CIN), and the result of the calculation is stored in the CRC data register (DCRA0COUT).

31.1.2 Block Diagram

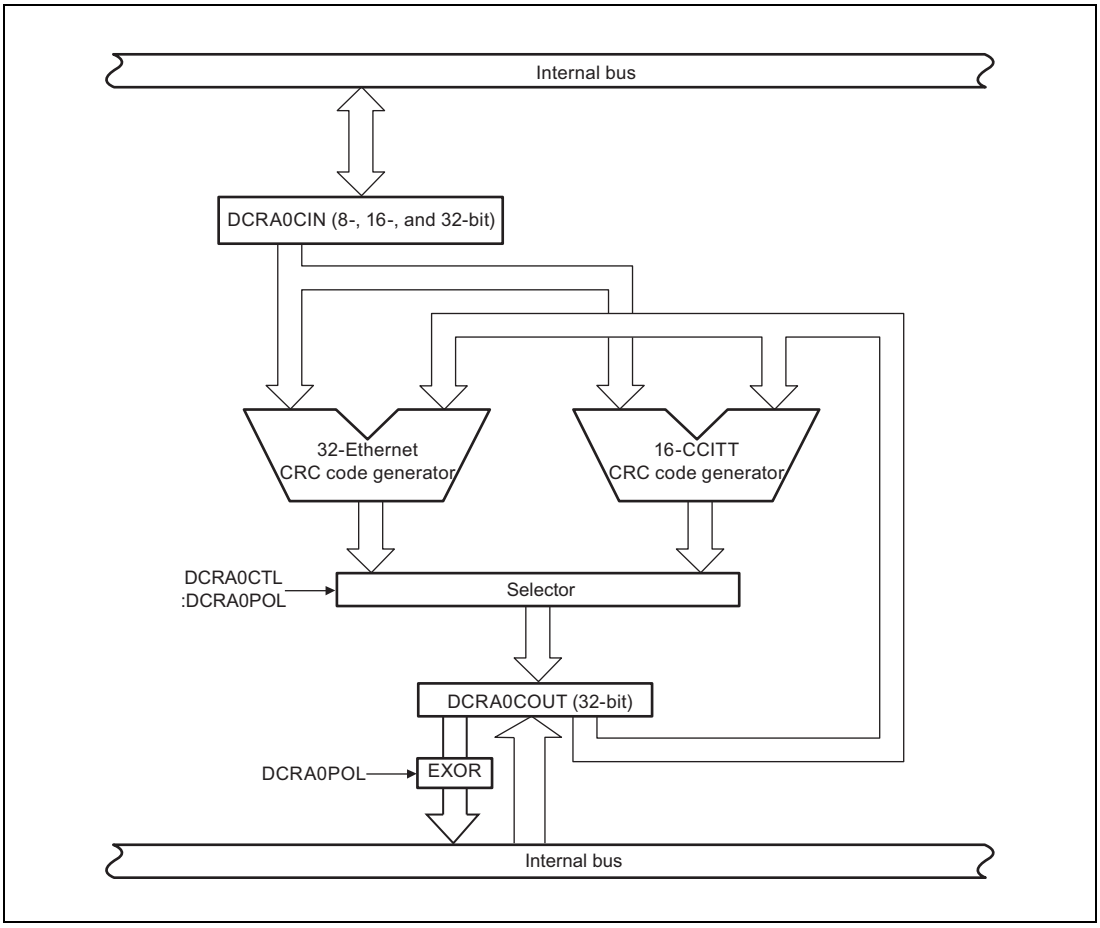
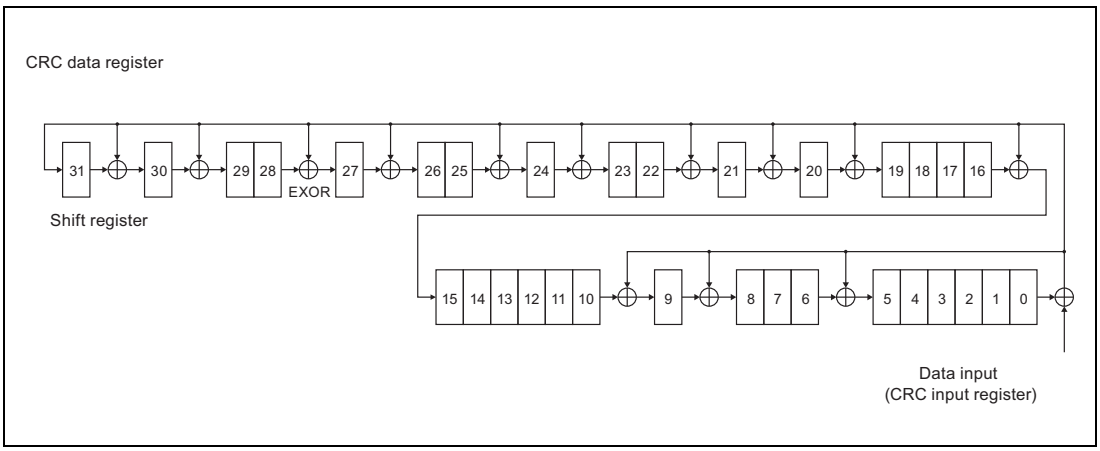


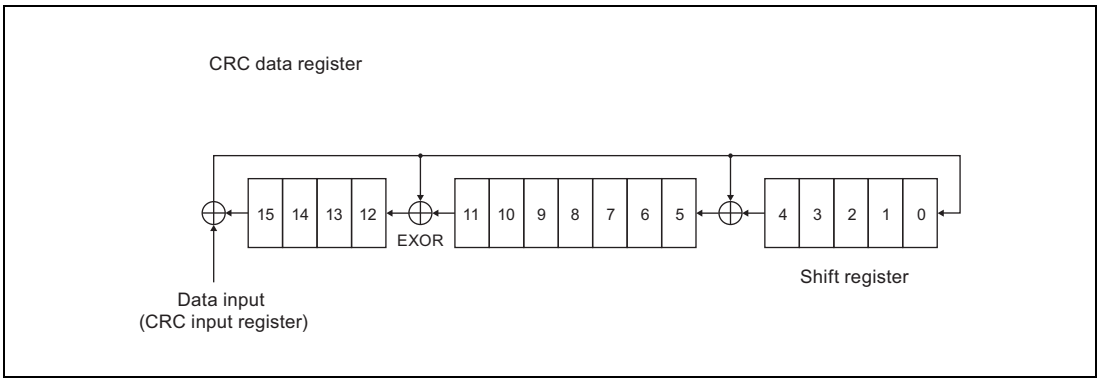
Figure 31.1 Data CRC Function A Block Diagram

31.1.3 Calculation Circuit

- 32-Ethernet



- 16-CCITT



31.2 Register Specifications

31.2.1 List of Registers

Table 31.2 List of DCRA Registers

Register Name	Function	Address
DCRA0CIN	CRC input register	<DCRA0_base> + 00_H
DCRA0COUT	CRC data register	<DCRA0_base> + 04_H
DCRA0CTL	CRC control register	<DCRA0_base> + 20_H

31.2.2 DCRA0CIN — CRC Input Register

The CRC input register is a 32-bit register that stores input data.

When data is set in this register, the CRC code corresponding to the input is generated.

The DCRA0ISZ1 and DCRA0ISZ0 bits of the CRC control register specify the effective bit width used in the CRC calculation circuit.

Access: This register can be read/written in 32-bit units.

Address: FFF7 0000$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DCRA0CIN															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DCRA0CIN															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 31.3 DCRA0CIN Register Contents

Bit Position	Bit Name	Function
31 to 0	DCRA0CIN [31:0]	Input Data for CRC Calculation The effective bit widths are: • For 32 bits: DCRA0CIN[31:0] • For 16 bits: DCRA0CIN[15:0] • For 8 bits: DCRA0CIN[7:0]

31.2.3 DCRA0COUT — CRC Data Register

The CRC data register is a 32-bit register that stores the result of CRC (32-Ethernet/16-CCITT) calculation.

Access: This register can be read/written in 32-bit units.

Address: FFF7 0004$_H$

Value after reset: 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	DCRA0COUT															
Value after reset *1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	DCRA0COUT															
Value after reset *1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Note 1. The read value after reset is 0000 0000$_H$ since the 32-bit Ethernet polynomial is selected as the CRC generating function after reset.

Table 31.4 DCRA0COUT Register Contents

Bit Position	Bit Name	Function
31 to 0	DCRA0COUT [31:0]	Result of the CRC Code Generation When the 16-bit CCITT polynomial is enabled, bits 15 to 0 show the CRC result. Bits 31 to 16 are undefined. The read value of this register is a value obtained by performing EXOR calculation for the following value: • For 32-bit Ethernet polynomial: FFFF FFFF$_H$ • For 16-bit CCITT polynomial: 0000$_H$ For example, when the 32-bit Ethernet polynomial is selected and DCRA0COUT is 5555 5555$_H$, AAAA AAAA$_H$ is read.

CAUTION

This register must be initialized (setting the initialized start value) before the first data of the data block is written to the DCRA0CIN register.

31.2.4 DCRA0CTL — CRC Control Register

The CRC control register is a bit register that controls CRC.

Access: This register can be read/written in 8-bit units.

Address: FFF7 0020_H

Value after reset: 00_H

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	—	DCRA0ISZ[1:0]		DCRA0POL
Value after reset	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R/W	R/W	R/W

Table 31.5 DCRA0CTL Register Contents

Bit Position	Bit Name	Function
7 to 3	—	Reserved
2,1	DCRA0ISZ[1:0]	Input Data Width Selection These bits specify the effective input data width of the CRC input register. bit2 bit1 0 0: Sets the effective input data width to 32 bits (DCRA0CIN31-0) 0 1: Sets the effective input data width to 16 bits (DCRA0CIN15-0) 1 0: Sets the effective input data width to 8 bits (DCRA0CIN7-0) 1 1: Setting prohibited
0	DCRA0POL	CRC Generating Polynomial Selection This bit specifies the CRC generating polynomial. 0: Selects 32-Ethernet (32-bit CRC code generation) 1: Selects 16-CCITT (16-bit CRC code generation)

CAUTIONS

1. **After changing the CRC generating function (DCRA0CTL.DCRA0POL), the DCRA0COUT register must be initialized.**
2. **The CRC bit width (DCRA0CTL.DCRA0ISZ[1:0]) must be set according to the data block bit width. Switching the CRC bit width is not allowed during processing of a data block. After the final CRC result is read from the DCRA0COUT register, the bit width can be changed. In this case, the DCRA0COUT register must be initialized (the initial start value must be set) afterwards.**

31.3 Functions

The data CRC function A generates a CRC (cyclic redundancy check) of an arbitrary data block length. The data is transferred to the data CRC function in 8-, 16- or 32-bit units. The CRC polynomial can either be selected for 32-bit Ethernet or 16-bit CCITT.

The initial start value must be set in the DCRA0COUT register before the first write access to the CRC input register (DCRA0CIN) is performed.

The flow chart below shows the CRC generating procedure.

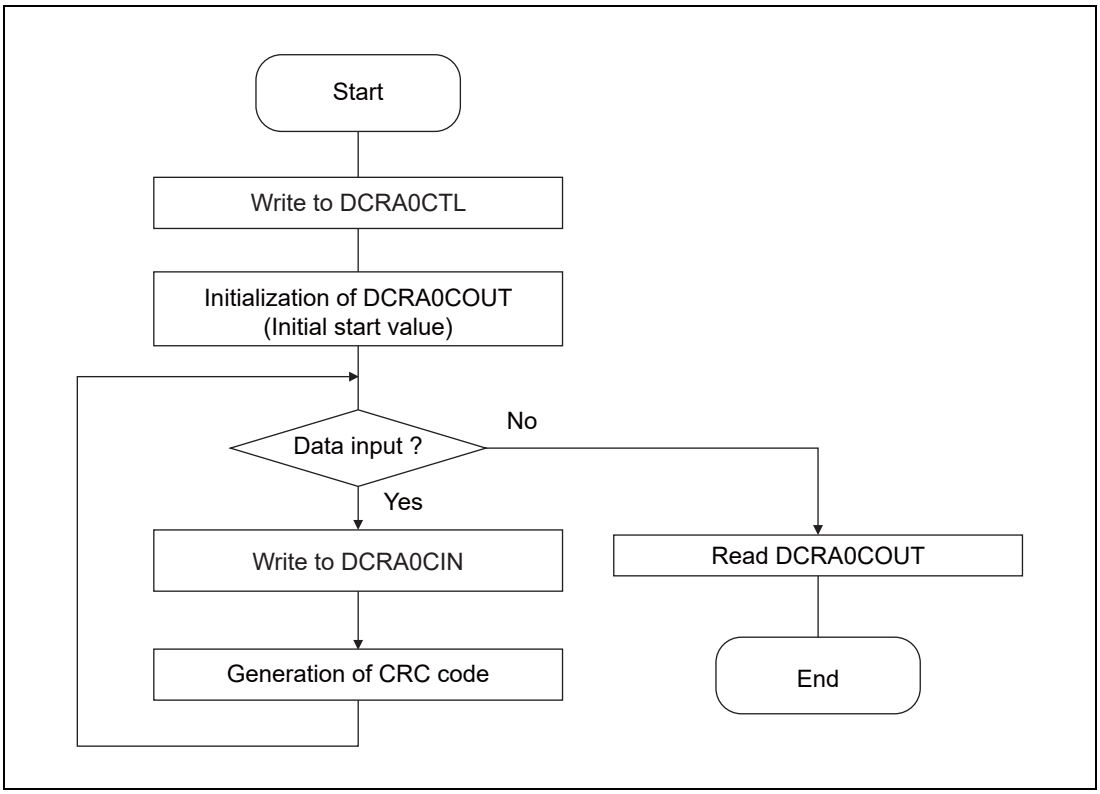


Figure 31.2 Flow of Data CRC Function A

NOTES

1. Before writing the first data to DCRA0CIN, the CRC output register DCRA0COUT must be initialized with the start value (initial start value).
2. Re-initialization of DCRA0COUT (initial start value) is also necessary when the polynomial is changed by changing DCRA0CTL.DCRA0POL.

Section 32 On-Chip Debugging Unit (OCD)

32.1 Debug Function

This microcontroller has an on-chip debug function. By using the on-chip debug emulator, programs can be debugged with the microcontroller mounted in the target system.

CAUTION

The debug functions described in this section are supported by the microcontroller but whether they are usable depends on the debugger. For details on debugging, see the user's manual of the debugger.

(1) Debug Interface

This microcontroller supports the NEXUS JTAG Interface, and low-pin debug interface (4-pin) (hereinafter referred to as LPD (4-pin)) as debug interfaces. Also it incorporates the AUD-RAM monitor and supports the AUD-RAM monitor interfaces to monitor and tune on-chip RAM data, peripheral registers, etc.

For the AUD-RAM monitor, see **Section 32.5, AUD-RAM Monitor (AUDR)**.

(2) Debug Monitoring

In debug mode, the monitoring program proceeds in the debugging-only area.

The basic debug functions below can be used by running a monitoring program.

- Downloading user programs
- Reading and writing user resources including the memory and registers while the user program is suspended
- Running the user program starting at any address

(3) On-chip Break Function

12 break points are included in the CPU1 and PCU each. Four of them can be designated for any access (access address and access data).

(4) Software Break Function

A software break point can be designated for any address.

(5) Forced Break Function

Execution of the user program can be forcibly suspended.

(6) Debug Interrupt Interface Function

Execution of the user program can be forcibly suspended by asserting an input signal on the $\overline{\text{EVTI}}$ pin from the outside.

(7) Forced Reset Function

The micro controller (this product) can be forcibly reset.

(8) Real-time RAM Monitoring (RRM)

The memory can be read during program execution. Because this read access uses debug-dedicated DMA, it has minimal effect on program execution.

(9) Dynamic Memory Modification (DMM)

The memory can be written during program execution. Because this write access uses debug-dedicated DMA, it has minimal effect on program execution.

(10) Timer Function

Using a 32-bit counter, the time for running the user program can be measured based on the clock for debug.

(11) Mask Function

A reset factor (external reset, software reset, and ECM reset) can be masked.

(12) Event Detection Function

Events can be detected by the following: execution address, access address, access data, range (comparison in size), and sequential execution.

(13) Trigger Input Interface

This microcontroller incorporates an event trigger input interface to accept external events. It can accept an external event in response to an input from the $\overline{\text{EVTI}}$ pin.

(14) Trigger Output Interface

An event trigger output interface is included in this microcontroller to notify the external debug device of event detection and the like. Output from the $\overline{\text{EVTO}}$ pin can output a trigger of event detection to the outside.

(15) Hot Plug-in Function

Debugging can be started in normal operating mode without an input of an external reset.

(16) Security Function

To prevent the contents of the flash memory from being read by an unauthorized person, a 128-bit ID code (OCD_ID) can be written to the microcontroller. If the code the user inputs when starting a debugger does not match the ID code written to the microcontroller, the flash memory cannot be accessed.

(17) Calibration Function

Emulating the flash memory and tuning ROM data can be executed by using the ERAM, the memory for emulation. For details, **Section 32.2, Calibration Function**.

(18) Tracing Function

Execution history, data changes, etc. of the user program can be obtained. For details, see **Section 32.3, Trace Control Function**.

(19) Multi Core Debug Function

The following functions are supported as the multi core debugger for the CPU1 and PCU: synchronization functions (including reset, execution, and break), synchronous setting.

32.2 Calibration Function

This microcontroller includes the emulation RAM as an emulation memory for the on-chip flash memory.

(1) Emulation RAM

An 8-Kbyte emulation RAM is provided.

(2) Flash Emulation Function

The emulation RAM can be mapped to any areas in the flash area .

- Bank 0: 8KB

(3) Tuning Function

The ROM data can be dynamically tuned during execution of the user program via the emulation RAM that has mapped to the flash area.

(4) Flash Cache Clear Function

The flash cache is cleared when the emulation RAM mapping is set. This preserves coherency of the contents between the on-chip flash memory or the emulation RAM and the flash cache memory.

32.3 Trace Control Function

This product can get the execution history of user program and changes of data by the software trace function.

The software information can be output through the debug interface (LPD (4-pin)).

32.4 Peripheral Break Control

The peripheral break function stops the peripheral modules if the user program is stopped, for instance upon a breakpoint hit.

The on-chip modules can be classified into two by its operation at the time of peripheral break as follows:

1. Modules that are unconditionally stopped: WDTA0, WDTA1
2. Modules that can select abeyance or continuation*1: OSTM0, OSTM1, OSTM2, TAPA2, TAPA3, TSG20, TSG21, ATU4

Note 1. When stopping the peripheral modules by peripheral break, TAPAn forcibly places TSG2n PWM output pins TSOn1 to TSOn6 in the Hi-Z state. Also, the value of the TAPAnHOF[10:8] bits in the TAPAnFLG register is set to 111_B.

32.5 AUD-RAM Monitor (AUDR)

32.5.1 Overview

This LSI includes the AUD (Advanced User Debugger) -RAM monitor (AUDR) to support debugging of a user program under conditions as if it were actually mounted in the system. The AUDR can read and write the resources mapped to memory spaces including an on-chip memory and a peripheral register during LSI operation.

Table 32.1 lists the outlines of the AUDR and **Figure 32.1** shows the block diagram of the AUDR.

Table 32.1 Outlines of the AUDR

Item	Outline
Transfer method	Clock-synchronous parallel interface (4 bits)
Transfer clock generation	Transfer clock is generated at the external host (RAM monitor tool) side.
Transfer clock frequency	Maximum 20 MHz
Access area	Physical address area on the system bus
Access data size	8, 16, 32, and 64 bits
Access address input bit width	8, 16, 24, and 32 bits The same values as those of the previous access address are used for the upper bits of access address, which are not input.
Data transfer method	• Single transfer: Single data is transferred to the access address that has been input. • Continuous transfer: Up to 16 data are continuously transferred from the access address that has been input. The access address is automatically incremented depending on data size.
I/O pin	7 pins ($\overline{\text{AUDRST}}$, AUDCK, $\overline{\text{AUDSYNC}}$, and AUDATA3 to AUDATA0)
Function	• RAM monitor function: Read and write are performed from a system bus to an accessible physical address area. This function enables reference to and modification of an on-chip memory, a peripheral register, and the like. • Configuration information retention (startup communication) function: The values of the AUDATA3 to AUDATA0 pins at the time of release from the internal reset state are retained. This function is used for communication with a RAM monitor tool. • Synchronization communication (message board) function: This flag register is used for communication of the firmware operated by the CPU with a RAM monitor tool.

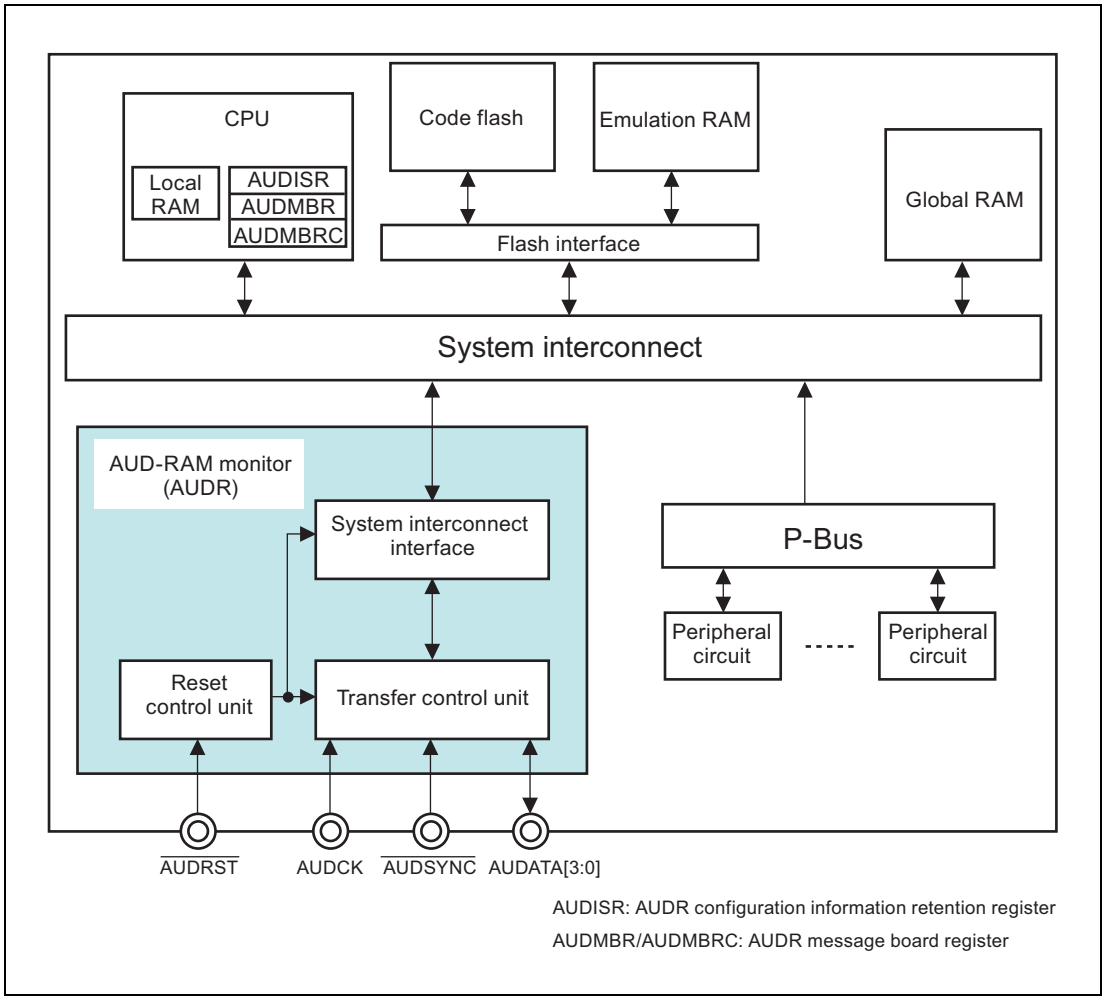


Figure 32.1 Blok Diagram of the AUDR

32.5.2 I/O Pins

Table 32.2 lists the I/O pins of the AUDR.

Table 32.2 I/O Pins of AUDR

Pin Name	I/O	Description
$\overline{\text{AUDRST}}$	Input	AUDR reset input pin. Inputting L to this pin resets the AUDR, but it does not initialize the AUDISR and AUDMBR/AUDMBRC (described below). When this pin is not connected, it is internally pulled-down.
AUDCK	Input	External clock input pin. The frequency of clock that can be input is less than 20 MHz. When this pin is not connected, it is internally pulled-up.
$\overline{\text{AUDSYNC}}$	Input	Timing control signal input pin. L: A command, an address, and write data are input and the status flag is output. H: The read-out data is output, and it causes the idle state. When this pin is not connected, it is internally pulled-up. **CAUTION** This pin should not be negated (to H) until the ready state is entered by inputting a command or the like from the outside to the AUDATA. For details, see the protocol mentioned below.
AUDATA3 to AUDATA0	Input/output	4-bit parallel data I/O pin. The following information is input and output by time division. • Command (input) • Address (input) • Write data (input)/Read data (output) • Status flag (output) When this pin is not connected, it is internally pulled-up.

32.5.3 Description of Registers

Table 32.3 lists the registers related to the AUDR.

These registers can be accessed only from CPU1.

Table 32.3 List of Registers

Register Name	Abbreviation	Value after Reset	R/W	Address	Access Size	Reference Section
AUDR configuration information retention register	AUDISR	$000X_H$*1	R	FA00 5000_H*4	16 *5	**32.5.3.1**
AUDR message board register	AUDMBR	0000_H	R/W *2	FA00 5004_H*4	16 *5	**32.5.3.2**
	AUDMBRC	0000_H	R/W *2,*3	FA00 5008_H*4	16 *5	

Note 1. The values of the AUDATA3 to AUDATA0 pins at the time of release from the internal reset state are retained in bit 3 to 0.

Note 2. Only 1 can be written to the bits that have been set to 0. 0 cannot be written to the bits that have been set to 1.

Note 3. All bits are cleared to 0 after read.

Note 4. The addresses of the AUDISR and AUDMBR/AUDMBRC bits differ when they are accessed from the program and the AUDR tool. Although these addresses are “Access prohibited” area, it is capable of accessing to this addresses for AUDR tool and communication.

Note 5. If an access in data sizes other than 16 bits (half-word) is performed, correct operation is not guaranteed.

32.5.3.1 AUDISR — AUDR Configuration Information Retention Register

AUDISR is a 16-bit readable register. It can be read out from the CPU and the AUDR.

Also it can be read out from the program and AUDR tool.

Use FA00 5000_H for an access from the program to the AUDISR.

Use F900 5000_H for an access from the AUDR tool to the AUDISR.

The AUDISR register is not initialized by AUDR reset. The AUDISR can be read out from the program during reset.

This register is used in the way described below.

- Retention of configuration information
  On release from an internal reset, the register retains the values corresponding to the levels on pins AUDATA3 to AUDATA0.

  Setting the levels on pins AUDATA3 to AUDATA0 in the emulator configuration allows programs to judge the connection configuration of the emulator.

  If nothing is connected to the AUD RAM monitoring-related pins, pins AUDATA3 to AUDATA0 are pulled up and the register is read as $000F_H$. If an AUD RAM monitoring tool is connected, the levels on pins AUDATA3 to AUDATA0 will correspond to a value other than $000F_H$. Reading of the AUDISR by the CPU can then be used to judge that a tool is connected.

  In addition, the values from pins AUDATA3 to AUDATA0 can be used to identify the vendor that is the source of the connected tool.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	DATA			
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	Values of the AUDATA3 to AUDATA0 pins			
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 32.4 AUDISR Register Contents

Bit Position	Bit Name	Function
15 to 4	—	Reserved These bits are always read as 0. The write value should always be 0.
3 to 0	DATA	The values of the AUDATA3 to AUDATA0 pins at the time of release from the internal reset state are retained.

32.5.3.2 AUDMBR/AUDMBRC — AUDR Message Board Register

The AUDMBR/AUDMBRC register is a 16-bit readable/writable register. It can be read and written from the program and the AUDR tool.

Use FA00 5004_H or FA00 5008_H for an access from the program. Use F900 5004_H or F900 5008_H for an access from the AUDR tool.

When the program reads the AUDMBRC at FA00 5008_H, all bits of the AUDMBR/AUDMBRC are cleared to 0 after reading. When the program reads the AUDMBR at FA00 5004_H, the bits of the AUDMBR/AUDMBRC are not cleared.

When the AUDR tool reads the AUDMBRC at F900 5008_H, all bits of the AUDMBR/AUDMBRC are cleared to 0 after reading. When the AUDR tool reads the AUDMBR at F900 5004_H, the bits of the AUDMBR/AUDMBRC are not cleared.

The program and the AUDR tool can write to the AUDMBR/AUDMBRC from the above addresses. However, writing 0 to the bits that have been set to 1 will be ignored (they can only be set to 1).

Table 32.6 summarizes the recommended accesses to the AUDMBR/AUDMBRC.

The AUDMBR/AUDMBRC register is not initialized by AUDR reset. The AUDMBR/AUDMBRC can be read and written from the program during AUDR reset.

AUDMBR and AUDMBRC are used in the way described below.

- Synchronous communications (message board) function
  The firmware (program) run by the CPU uses AUDMBR as a flag register for use in communications with the emulator, which reads AUDMBR by using the RAM monitoring function. This allows the emulator to follow the state of firmware (program) operations.

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	AUDMBR															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 32.5 AUDMBR/AUDMBRC Register Contents

Bit Position	Bit Name	Function
15 to 0	AUDMBR	Communication flag between the AUDR and the CPU

Table 32.6 Recommended Access to the AUDMBR/AUDMBRC

Accessing Master	Address	R/W	Access	Remarks
Program	FA00 5004_H (AUDMBR)	Write	Only 1 can be written. Writing 0 is ignored.	—
		Read	Reading is enabled.	Not cleared after reading.
AUDR tool	F900 5008_H (AUDMBRC)	Write	Only 1 can be written. Writing 0 is ignored.	—
		Read	Reading is enabled.	All bits are cleared to 0 after reading.

32.5.4 RAM Monitoring

32.5.4.1 Communication Protocol

Input a command, counter value, address, and data to the AUDATA pin in the format shown in **Figure 32.2**. For details, see **Section 32.5.4.2, Operation**.

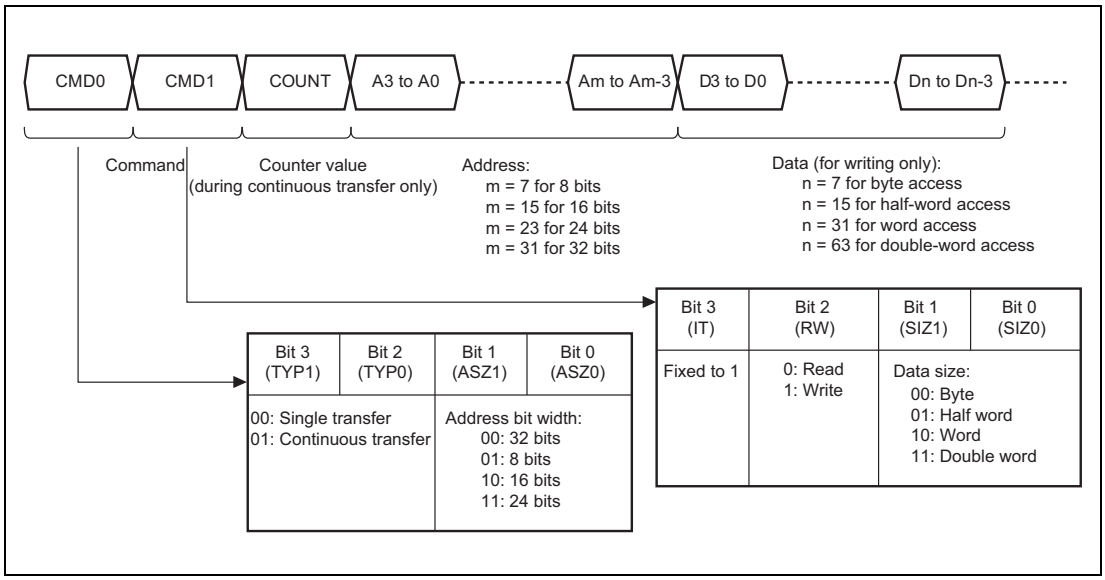


Figure 32.2 AUDATA Pin Input Format

32.5.4.2 Operation

(1) Single Transfer

Single transfer is a method of transmitting single data to an address that has been input for access. **Figure 32.3** shows an example of reading operation during single transfer and **Figure 32.4** shows an example of writing operation during single transfer.

When the $\overline{\text{AUDSYNC}}$ pin is asserted and a command, address and data (only for writing) are input to the AUDATA pin in the format shown in **Figure 32.2**, the AUDR starts reading or writing the specified address. The AUDR outputs the Not Ready flag (0000) to the AUDATA pin during this internal execution. After successful completion of the internal execution, the AUDR outputs the Ready flag (0001) to the AUDATA pin. (See **Figure 32.3** and **Figure 32.4**).

For reading, after the Ready flag is output and the $\overline{\text{AUDSYNC}}$ pin is negated, the read data is output from the AUDATA pin (see **Figure 32.3**).

After output of the read data and until input of the next command, at least one AUDCK cycle is required for switching of the input/output states of the pins.

In addition, after the $\overline{\text{AUDSYNC}}$ pin is negated following the completion of processing for write access, when the next command is input, at least one AUDCK cycle is required for switching of the input/output states of the pins. Therefore, the $\overline{\text{AUDSYNC}}$ pin is negated for two cycles of AUDCK.

If a command other than those shown in **Figure 32.2** is input to CMD0 and CMD1, the AUDR disables this processing as a command error and sets the CFLG bit in the Ready flag to 1. When a bus error occurs during the internal execution, the AUDR disables this processing and sets the BFLG bit in the Ready flag to 1 (see **Figure 32.5**). When an error is detected, the read data is not output.

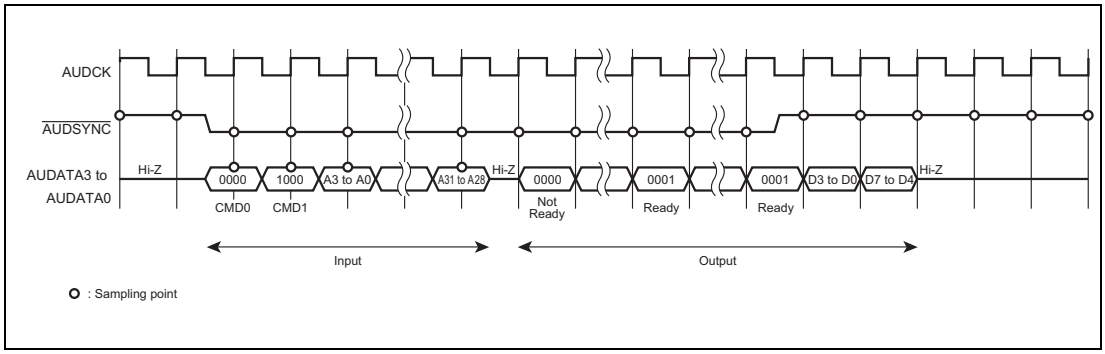


Figure 32.3 Example of Reading Operation in Single Transfer (Address 32 Bits; Byte Read)

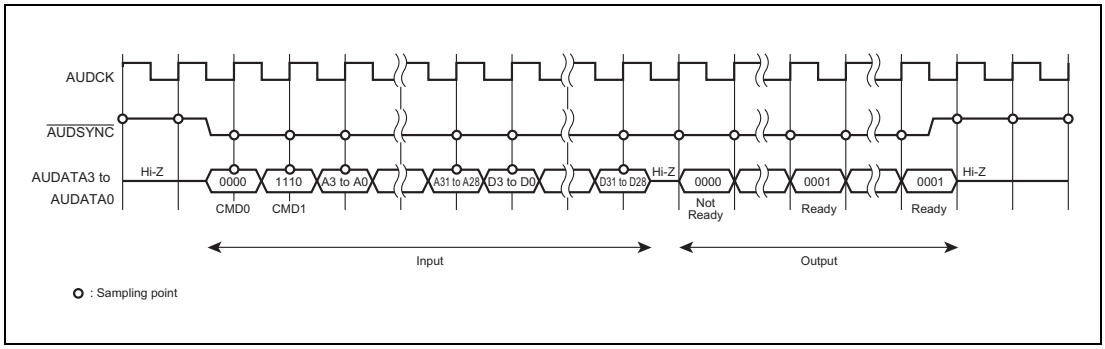


Figure 32.4 Example of Writing Operation in Single Transfer (Address 32 Bits; Word Write)

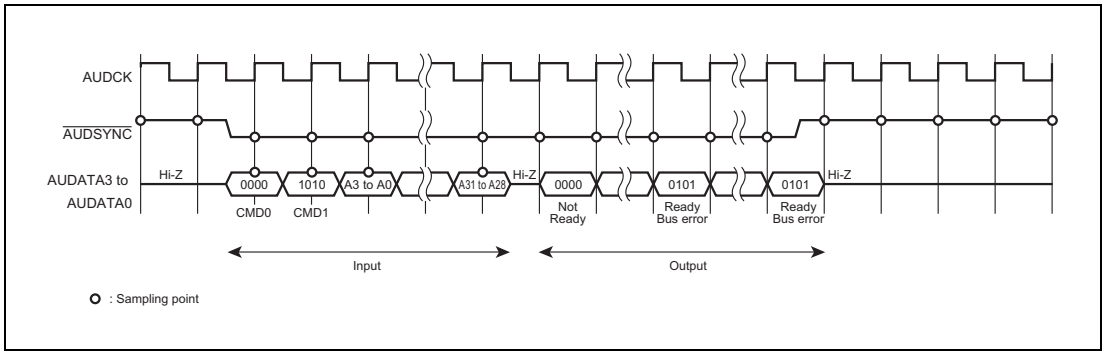


Figure 32.5 Example of Error Occurrence in Single Transfer (Address 32 Bits; Word Read)

(2) Continuous Transfer

Continuous transfer is a method of continuously transmitting up to 16 data to the address that has been input. The address to be accessed is automatically incremented depending on the data size after completion of each data transfer. **Figure 32.6** shows an example of reading operation during continuous transfer and **Figure 32.7** shows an example of writing operation during continuous transfer.

The first data transfer is equivalent to single transfer except that COUNT (counter value) shown in **Figure 32.2** should be input. The number of data to be transferred minus 1 is input to the COUNT.

In the second and subsequent data transfer, the input of CMD0, CMD1, COUNT, and addresses is skipped. The second and subsequent data transfer for reading and writing proceed as follows.

For reading, when the $\overline{\text{AUDSYNC}}$ pin is asserted after completion of the previous data transfer, the AUDR outputs the Not Ready flag (0000) to the AUDATA pin during this internal execution. After successful completion of the internal execution, the AUDR outputs the Ready flag (0001) to the AUDATA pin. Then after the $\overline{\text{AUDSYNC}}$ pin is negated, the read data is output from the AUDATA pin. This operation is repeated until the number of data specified in COUNT is read (see **Figure 32.6**).

For writing, when the $\overline{\text{AUDSYNC}}$ pin is asserted after completion of the previous data transfer, the AUDR inputs data to be written to the AUDATA pin. After that, the AUDR starts the internal execution. The AUDR outputs the Not Ready flag (0000) to the AUDATA pin during this operation. After successful completion of the internal execution, the AUDR outputs the Ready flag (0001) to the AUDATA pin. This operation is repeated until the number of data specified in the COUNT is written (see **Figure 32.7**).

When the $\overline{\text{AUDSYNC}}$ pin is re-asserted to input the next write data, at least two cycles of AUDCK are required for negating the $\overline{\text{AUDSYNC}}$ pin.

If a bus error occurs during the internal execution, the AUDR suspends data transfer and sets the BFLG bit in the Ready flag to 1 (see **Figure 32.8**).

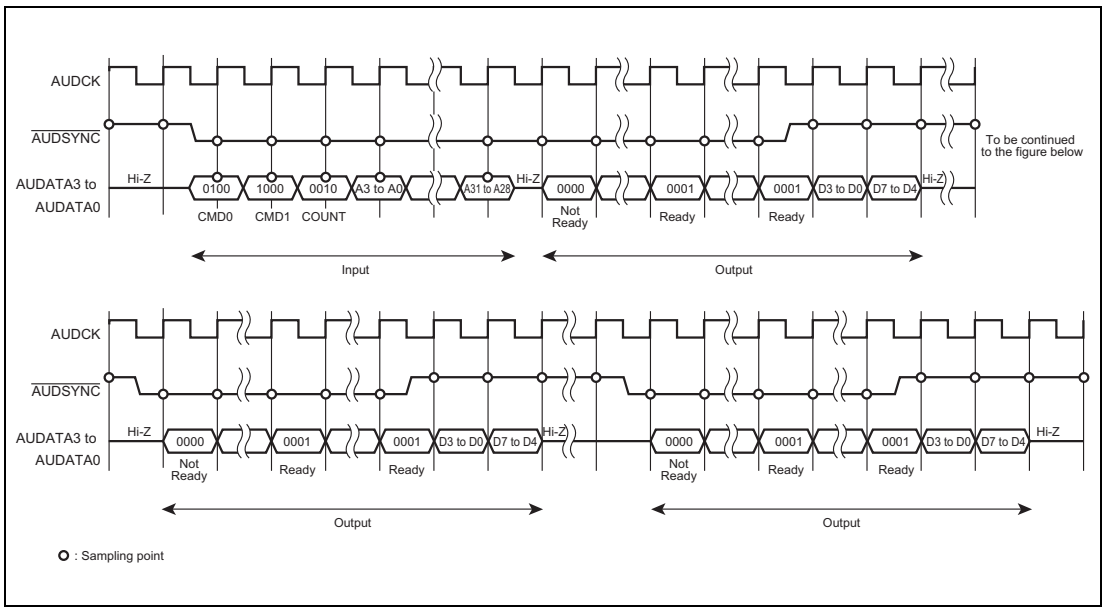


Figure 32.6 Example of Reading Operation in Continuous Transfer (Address 32 Bits; Byte Read × Three Units of Data)

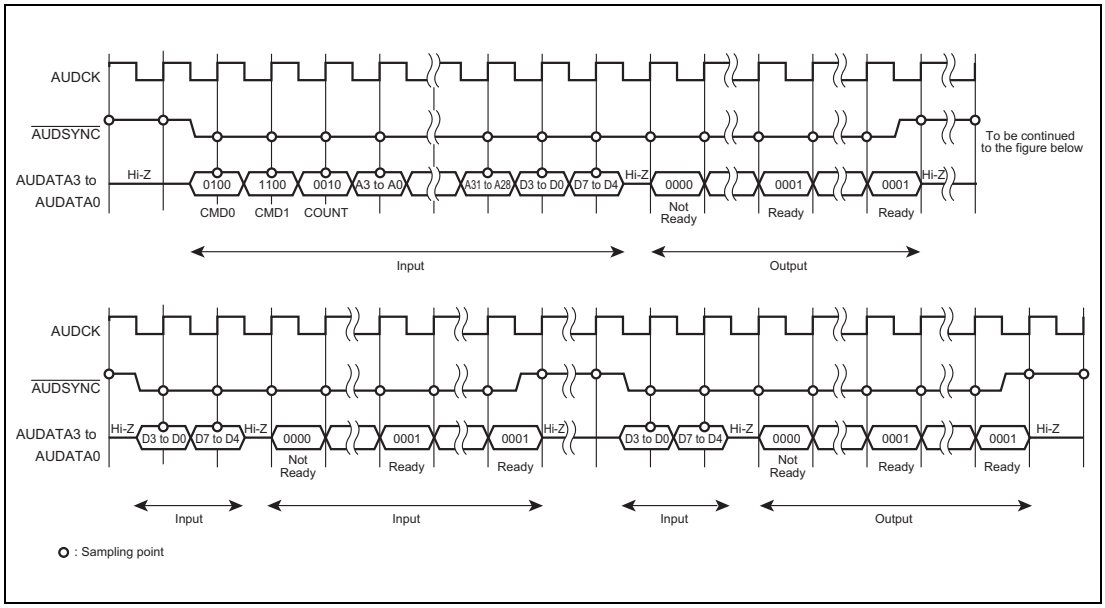


Figure 32.7 Example of Writing Operation in Continuous Transfer (Address 32 Bits; Byte Write × Three Units of Data)

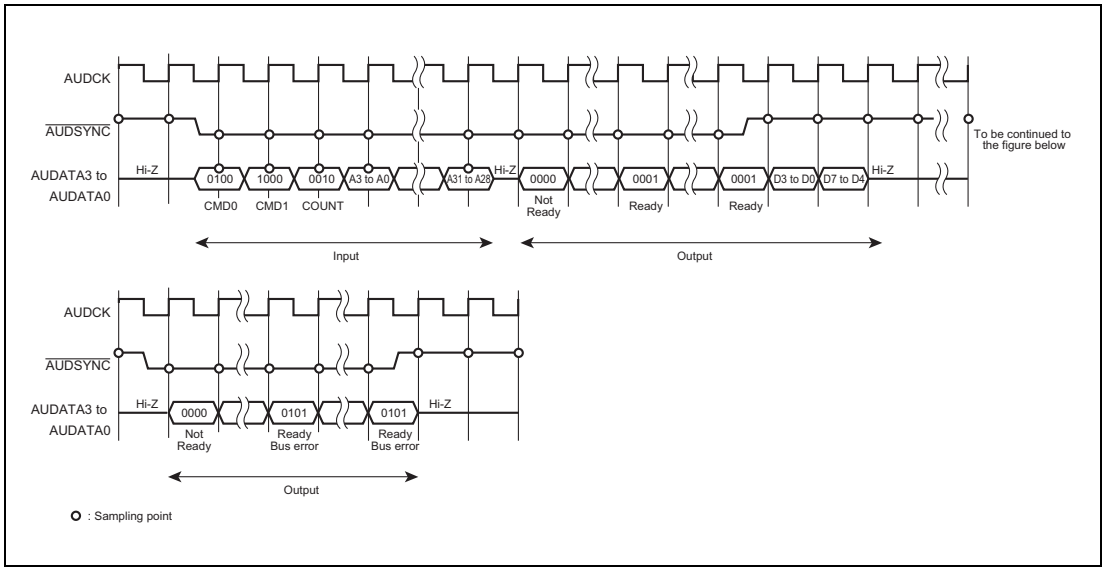


Figure 32.8 Example of Error Occurrence in Continuous Transfer (Address 32 Bits; Byte Read × Three Units of Data)

(3) Command Error Conditions

Table 32.7 Command (CMD0) Error Conditions

Bit 3 (TYP1)	Bit 2 (TYP0)	Bit 1 (ASZ1)	Bit 0 (ASZ0)	Description
0	0	0	0	Single transfer Address bit width: 32 bits
0	0	0	1	Single transfer Address bit width: 8 bits
0	0	1	0	Single transfer Address bit width: 16 bits
0	0	1	1	Single transfer Address bit width: 24 bits
0	1	0	0	Continued transfer Address bit width: 32 bits
0	1	0	1	Continued transfer Address bit width: 8 bits
0	1	1	0	Continued transfer Address bit width: 16 bits
0	1	1	1	Continued transfer Address bit width: 24 bits
1	×	×	×	Command error

Table 32.8 Command (CMD1) Error Conditions

Bit 3 (IT)	Bit 2 (RW)	Bit 1 (SIZ1)	Bit 0 (SIZ0)	Description
0	×	×	×	Command error
1	0	0	0	Read: Byte
1	0	0	1	Read: Half word
1	0	1	0	Read: Word
1	0	1	1	Read: Double word
1	1	0	0	Write: Byte
1	1	0	1	Write: Half word
1	1	1	0	Write: Word
1	1	1	1	Write: Double word

(4) Bus Error Conditions

- Half-word access is made to the address of 4n + 1 and 4n + 3.
- Word access is made to the address of 4n + 1, 4n + 2, and 4n + 3.
- Double-word access is made to the address of 8n + 1, 8n + 2, 8n + 3, 8n + 4, 8n + 5, 8n + 6, and 8n + 7.
- An error response is received from the system bus.

(5) AUDATA Pin Input Format

Table 32.9 Input Format Bit Position

Input Order	Format Name	Bit Position AUDATA 3	AUDATA 2	AUDATA 1	AUDATA 0	√: Required —: Not Required			
Earlier	CMD0	TYP1	TYP0	ASZ1	ASZ0	√			
	CMD1	IT	RW	SIZ1	SIZ0	√			
	COUNT	C3	C2	C1	C0	—: Single transfer; √: Continuous transfer			
	Address					8 bits	16 bits	24 bits	32 bits
		A3	A2	A1	A0	√	√	√	√
		A7	A6	A5	A4	√	√	√	√
		A11	A10	A9	A8	—	√	√	√
		A15	A14	A13	A12	—	√	√	√
		A19	A18	A17	A16	—	—	√	√
		A23	A22	A21	A20	—	—	√	√
		A27	A26	A25	A24	—	—	—	√
		A31	A30	A29	A28	—	—	—	√
	Data (for writing only)					Byte write	Half-word write	Word write	Double-word write
		D3	D2	D1	D0	√	√	√	√
		D7	D6	D5	D4	√	√	√	√
		D11	D10	D9	D8	—	√	√	√
		D15	D14	D13	D12	—	√	√	√
		D19	D18	D17	D16	—	—	√	√
		D23	D22	D21	D20	—	—	√	√
		D27	D26	D25	D24	—	—	√	√
		D31	D30	D29	D28	—	—	√	√
		D35	D34	D33	D32	—	—	—	√
		D39	D38	D37	D36	—	—	—	√
		D43	D42	D41	D40	—	—	—	√
		D47	D46	D45	D44	—	—	—	√
		D51	D50	D49	D48	—	—	—	√
		D55	D54	D53	D52	—	—	—	√
		D59	D58	D57	D56	—	—	—	√
Later		D63	D62	D61	D60	—	—	—	√

Table 32.10 CMD0 Format

Bit Name	Function	Description
TYP[1:0]	Type of transfer	00: Single transfer 01: Continuous transfer
ASZ[1:0]	Specify an address bit width	This bit specifies the bit width of an address input from the AUDATA pin. When 8, 16, or 24 bits are specified, the same values as the previous access address are used for the upper bits that are not input from the AUDATA pin. Input a 32-bit address for the first access after reset release or occurrence of a command or bus error. 00: 32 bits 01: 8 bits 10: 16 bits 11: 24 bits

Note: Settings other than the above cause a command error.

Table 32.11 CMD1 Format

Bit Name	Function	Description
IT	Specifies access space	Set this bit to 1.
RW	Specifies read or write	0: Read 1: Write
SIZ[1:0]	Specify data size	These bits specify the size of data to be accessed. 00: Byte (8 bits) 01: Half word (16 bits) 10: Word (32 bits) 11: Double word (64 bits)

Note: Settings other than the above cause a command error.

Table 32.12 COUNT Format

Bit Name	Function	Description
C3 to C0	Specify the number of data to be transferred	These bits specify the number of data to be transferred in continuous transfer. 0000: 1 data 0001: 2 data 0010: 3 data 0011: 4 data 0100: 5 data 0101: 6 data 0110: 7 data 0111: 8 data 1000: 9 data 1001: 10 data 1010: 11 data 1011: 12 data 1100: 13 data 1101: 14 data 1110: 15 data 1111: 16 data

Table 32.13 Address Format

Bit Name	Function	Description
A31 to A0	Specify address	These bits specify an address to be accessed. The required number of bits depends on the setting of the ASZ[1:0] bits in CMD0 (for details, see **Table 32.9**).

Table 32.14 Write Data Format

Bit Name	Function	Description
D63 to D0	Specify write data	These bits specify write data. The required number of bits depends on the setting of the SIZ[1:0] bits in CMD1 (for details, see **Table 32.9**).

(6) AUDATA Pin Output Format

Table 32.15 Ready Flag Format

Bit Position	Bit Name	Function	Description
AUDATA3	0	—	—
AUDATA2	BFLG	This bit indicates a bus error.	0: Normal 1: A bus error occurred.
AUDATA1	CFLG	This bit indicates a command error.	0: Normal 1: A command error occurred.
AUDATA0	RFLG	This bit indicates completion of AUDR operation.	0: Not Ready 1: Ready

Table 32.16 Read Data Bit Position

	Bit Position				√: Required; —: Not Required			
Output Order	AUDATA3	AUDATA2	AUDATA1	AUDATA0	Byte read	Half-word read	Word read	Double-word read
Earlier	D3	D2	D1	D0	√	√	√	√
	D7	D6	D5	D4	√	√	√	√
	D11	D10	D9	D8	—	√	√	√
	D15	D14	D13	D12	—	√	√	√
	D19	D18	D17	D16	—	—	√	√
	D23	D22	D21	D20	—	—	√	√
	D27	D26	D25	D24	—	—	√	√
	D31	D30	D29	D28	—	—	√	√
	D35	D34	D33	D32	—	—	—	√
	D39	D38	D37	D36	—	—	—	√
	D43	D42	D41	D40	—	—	—	√
	D47	D46	D45	D44	—	—	—	√
	D51	D50	D49	D48	—	—	—	√
	D55	D54	D53	D52	—	—	—	√
	D59	D58	D57	D56	—	—	—	√
Later	D63	D62	D61	D60	—	—	—	√

Table 32.17 Read Data Format

Bit Name	Function	Description
D63 to D0	Output read data	The required number of bits depends on the setting of the SIZ[1:0] bits in CMD1 (for details, see **Table 32.16**).

32.5.4.3 Usage Notes on AUDR Function

- Do not negate the $\overline{\text{AUDSYNC}}$ pin until one cycle of AUDCK has elapsed after a command is input to the AUDATA pin and the Ready flag has been returned.
- When uninitialized memory is accessed through the AUDR, a bus error may occur due to ECC error detection.

32.5.4.4 Enabling and Disabling the RAM Monitor Function

Enabling or disabling of the AUDR can be specified using option bytes (stored in the flash memory and set by reset transfer).

AUDREN: AUDR Enable Bit

For setting of option bytes, see **Section 33, Flash Memory**.

NOTE

In serial programming mode, AUDR is disabled regardless of this setting.

32.6 Caution on Using On-Chip Debugger

(1) Treatment of Devices Used for Debugging
Do not install a device that was used for debugging on a mass-produced product, because the flash memory was rewritten during system debugging and thus the number of write/erase count of flash memory cannot be guaranteed.

(2) This product does not support hot plug out for power off (including removal of the connector) of the debug tool during debug mode. Do not turn off the power of the NEXUS tool (including removal of the connector) in debug mode.

(3) When ending on-chip debugging, set the $\overline{\text{TRST}}$ pin and external reset pin to the low level.

(4) When a debugger is used, the program written to the microcomputer before preparation for communications between the OCD emulator and microcomputer is complete is executed from the reset vector. For this reason, take care to ensure that this does not lead to any operations which you do not expect or intend.
This communications preparation period depends on the host PC environment of the OCD emulator and the operating frequency of the microcomputer.

Section 33 Flash Memory

This product incorporates code flash memory and data flash memory.

33.1 Features

- Code flash memory capacity: 4 Mbytes of user area and 32 Kbytes of user boot area
- Data flash memory capacity: 64 Kbytes of data area
- Methods of programming
  - Programming through transfer by a dedicated flash-memory programmer via a serial interface communication (serial programming)
  - Programming of flash memory by a user program (self-programming)
- Support for security functions to protect against illicit tampering and illicit reading with data in flash memory
- Support for protection functions to protect against erroneous overwriting of the flash memory
- Support for the detection and correction of errors in the flash memory
- Support for the BGO (Back Ground Operation) function
  - Code flash memory can be read while data flash memory is being programmed
- Interrupts acceptable during self-programming
- The initial settings of this product can be configured in the extended area (option bytes) of flash memory.

33.2 Structure of Memory

Figure 33.1 shows code flash memory map.

The user area in the code flash memory of this product is divided into 8- and 32-Kbyte blocks, which can be erased individually. The user area can be used to hold the user program. The user boot area can be used to hold a non-rewritable boot program during user program operation, such as a boot program for rewriting code flash memory through a selected user interface.

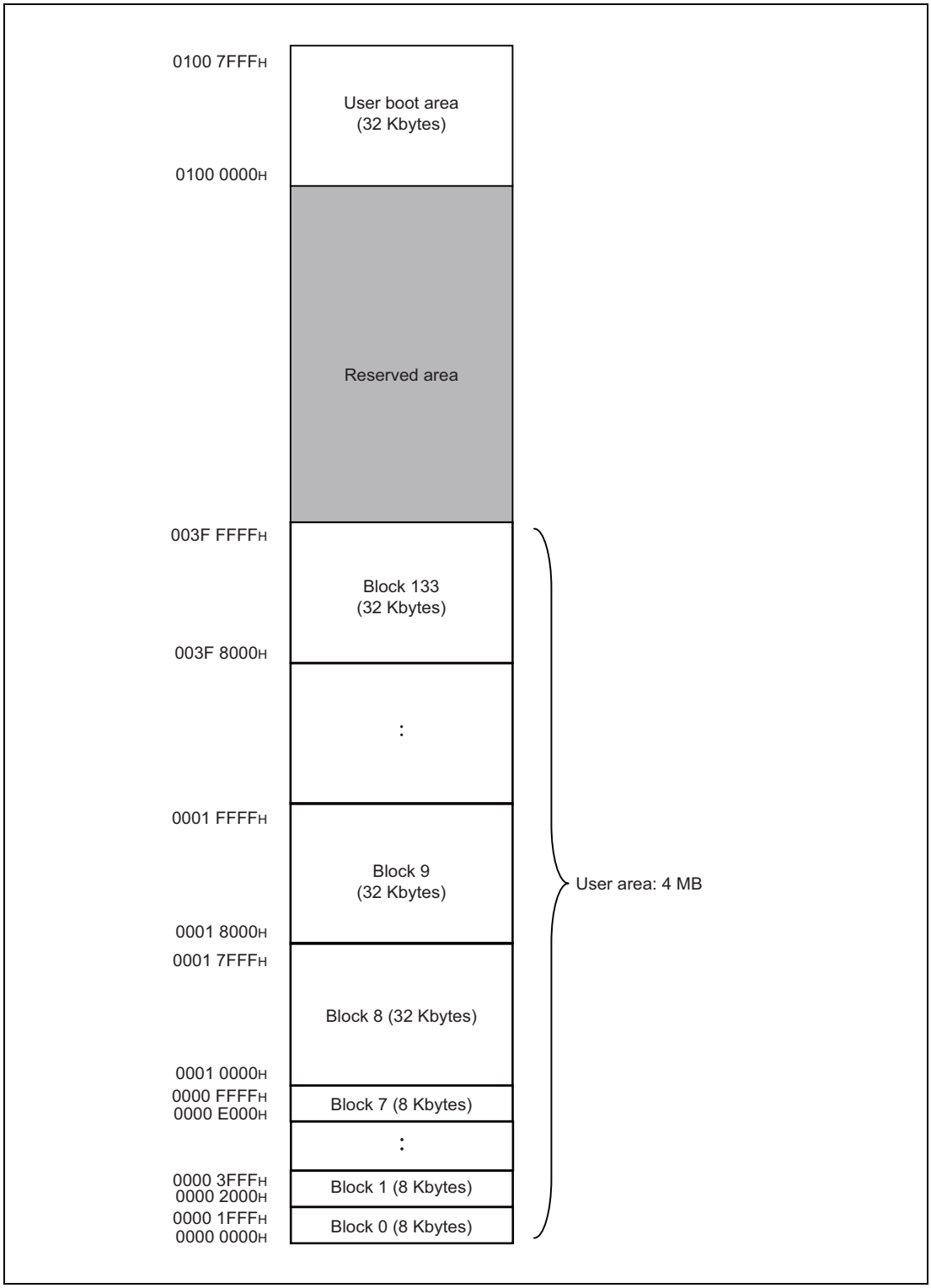


Figure 33.1 Code Flash Memory Map (1 bank, 8 Kbytes × 8 + 32 Kbytes × 126 configuration)

The data area of the data flash memory in this product is divided into 64-byte blocks, with each being a unit for erasure. **Figure 33.2** shows data flash memory map.

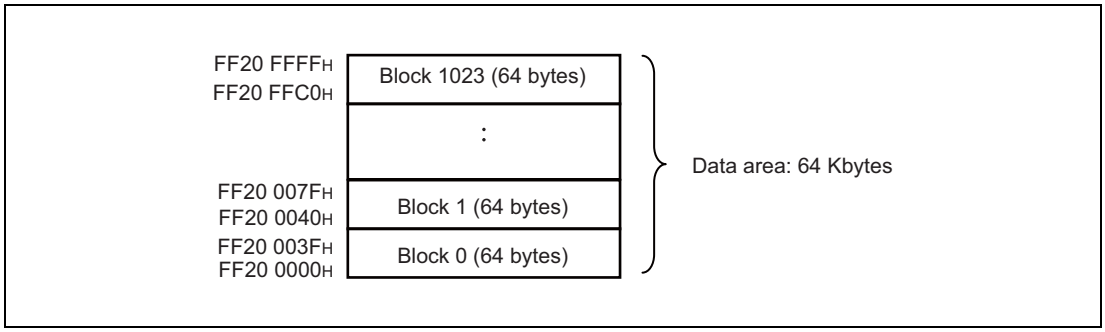


Figure 33.2 Data Flash Memory Map

33.3 Operating Modes Associated with Flash Memory

Figure 33.3 is a diagram of the mode transitions associated with the flash memory. For the procedures for setting the modes, refer to **Section 5, Operating Mode**.

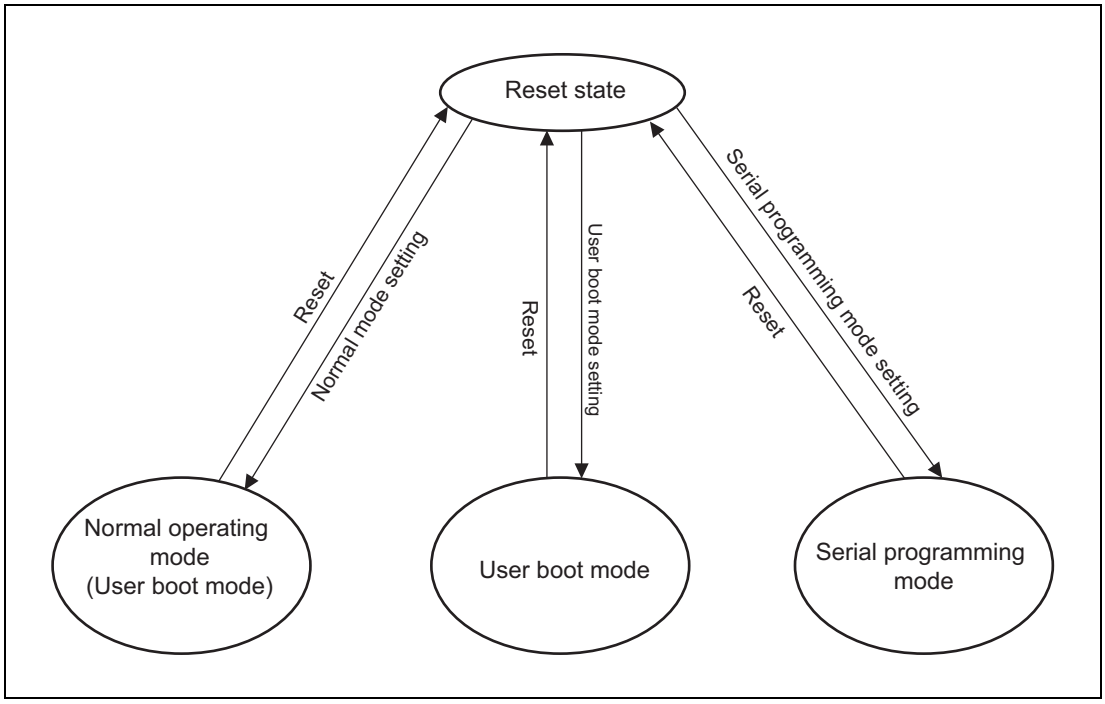


Figure 33.3 Mode Transition Associated with Flash Memory

The flash memory areas which are programmable and erasable and the boot program after a reset depend on the selected mode. The differences between modes are indicated in **Table 33.1**.

Table 33.1 Differences between Modes

Item	Normal Operating Mode (User boot mode)*1	User Boot Mode	Serial Programming Mode
Programmable and erasable area	• User area • Data area	• User area • Data area	• User area • User boot area • Data area
Boot program at a reset	Program in user area	Program in user boot area	Embedded program for serial programming

Note 1. See **Section 5 Operating Mode** for the details on each mode.

33.4 Functional Overview

By using a dedicated flash-memory programmer to program the flash memory of this device via a serial interface (serial programming), the device can be programmed regardless of whether this is before or after it is mounted on the target system.

Furthermore, security functions to prohibit updating of the user program written in the on-chip flash memory are incorporated, and this can prevent tampering by third parties.

Programming by the user program (self-programming) is suited for applications where the target system program may require updating after the production or shipment. Protection features for the safe rewriting of the flash memory area are also incorporated. Furthermore, interrupt processing during self-programming is supported, so programming can proceed at the same time as interrupt processing on an external communication control, etc., and this allows programming under various conditions.

Table 33.2 gives an overview of the methods of programming and the corresponding operating modes.

Table 33.2 Programming Methods

Programming Method	Functional Overview	Operating Mode
Serial programming	A dedicated flash-memory programmer is capable of on-board programming the flash memory after the device is mounted on the target system.	Serial programming mode
	A dedicated flash memory programmer and dedicated programming adapter board allow off-board programming of the flash memory, i.e. programming of the device before it is mounted on the target system.	
Self-programming	The user program that is written to code flash memory in advance by serial programming executing also allows updating the flash memory. The background operation capability makes it possible to fetch instructions or otherwise read data in code flash memory while the data flash memory is being programmed. For this reason, it is possible to update the data flash memory by executing a program written to the code flash memory. Instructions in the code flash memory cannot be fetched and data cannot be read while the code flash memory is being updated by self-programming. In such cases, a program for updating must be transferred to the local RAM or the global RAM in advance and executed.	Normal operating mode, user boot mode

When executing self-programming, see the *RH850/E1x Flash Memory User's Manual: Hardware Interface* which this product targets.

Table 33.3 lists the functions of the on-chip flash memory. Dedicated flash memory programmer commands realize serial programming, while reading of the on-chip flash memory by interface operation of flash memory or the user program realizes self-programming.

Table 33.3 Basic Functions at a Glance

Function	Description in Overview	Level of Support (√: Supported, △: Conditionally Supported, ×: Not Supported)	
		Serial programming	Self-programming
Blank checking	This is used to check a specified block to ensure that writing to it has not already proceeded. Results of reading from code flash memory and data flash memory to which nothing has been written after erasure are not guaranteed, so use blank checking to confirm that writing to memory has not proceeded after erasure.	√	√
Block erasure	This is for erasing the contents of a specified block of memory.	√	√
Programming	This is for writing to a specified address.	√	√
Verification and checksum	Data that are read out from flash memory are compared with data transferred from the flash memory programmer.	√	√
Reading	Data that have been written to the flash memory are read out.	√	√
Setting for OTP (one-time programming)	A specified block of code flash memory is set for OTP (OTP can only be set, that is, it is not possible to release a block's OTP setting).	√	√
Setting an ID	An ID setting is made for use in controlling the connection of a dedicated flash memory programmer for serial programming, and programming of the code flash memory by self-programming.	√	√
Security settings	Security settings are for use in serial programming.	√	△ (only when setting is prohibited after being permitted)
Protection settings	Lock bits for all blocks of code flash memory are provided.	√	√
Setting of option bytes	Option bytes are set to change them from the initial values for this product.	√	√
Clearing the configuration	ID setting, security settings, protection settings, and option byte settings are initialized.	√	×

For details on serial programming, refer to the *PG-FP5 Flash Memory Programmer User's Manual* and *Renesas Flash Programmer Flash Programming Software User's Manual.*

For details on self-programming, refer to the *RH850/E1x Flash Memory User's Manual: Hardware Interface* which this product targets.

The on-chip flash memory supports various security functions.

The OTP setting and authentication of the ID code are security functions for use with serial programming and self-programming.

In serial programming, authentication of the ID code, prohibiting connection of a dedicated flash memory programmer, and prohibition of commands (for block erasure, programming, and reading) are available for use as security functions.

Table 33.4, Summary of Security Functions lists security functions that are supported by the on-chip flash memory and **Table 33.5, Available Operations and Security Settings** lists security setting operations.

Table 33.4 Summary of Security Functions

Function	Description
OTP	OTP can be individually set for each block of the user area and the user boot area of code flash memory. When the OTP setting is made for an area, programming by serial programming and by self programming is prohibited. Once set, the OTP setting cannot be released. Furthermore, since execution of the configuration clearing command is prohibited for any area for which OTP has been set, changing a security setting from "prohibited" to "permitted" is not possible.
ID authentication	The result of ID authentication can be used to control the connection of a dedicated flash memory programmer for serial programming. The result of ID authentication can also be used to control enabling of code flash memory writing by self-programming.
Prohibition of connection of a dedicated flash memory programmer	The connection of a dedicated flash memory programmer for serial programming is prohibited. Since execution of the configuration clearing command is also prohibited when the connection of a dedicated Flash Memory programmer is prohibited, changing a security setting from "prohibited" to "permitted" is not possible.
Prohibition of block erasure commands	Block erasure commands at the time of serial programming are prohibited. Since execution of the configuration clearing command is also prohibited when block erasure commands are prohibited, changing a security setting from "prohibited" to "permitted" is not possible.
Prohibition of programming commands	Block erasure commands and programming commands at the time of serial programming are prohibited. Block erasure commands can only be executed by erasing all user areas in the following order starting from block 0: erasure of all areas → erasure of user boot area → erasure of data areas starting from block 0. The prohibition setting can only be initialized by issuing the configuration clearing command.
Prohibition of read commands	Read commands at the time of serial programming are prohibited. Only through execution of the configuration clearing command can the prohibition be lifted.

Table 33.5 Available Operations and Security Settings

Function	All Security Settings and Erasure, Programming, and Read Operations (√: Executable, ×: Not Executable, —: Not Supported)		Point for Caution Regarding the Security Setting	
	Serial programming	Self programming	Serial programming	Self programming
OTP	• Areas for which OTP is set – Block erasure commands: × – Programming commands: × – Read commands: √ • Areas for which OTP is not set – Block erasure commands: √ – Programming commands: √ – Read commands: √	• Areas for which OTP is set – Block erasure: × – Programming: × – Reading: √ • Areas for which OTP is not set – Block erasure: √ – Programming: √ – Reading: √	• The OTP setting cannot be released. • Execution of the configuration clearing command is not possible.	The OTP setting cannot be released.
ID authentication	• When the ID codes do not match – Block erasure commands: × – Programming commands: × – Read commands: × • When the ID codes match – Block erasure commands: √ – Programming commands: √ – Read commands: √	• When the ID codes do not match – Code flash memory - Block erasure: × - Programming: × - Reading: √ – Data flash memory - Block erasure: √ - Programming: √ - Reading: √ • When the ID codes match – Block erasure: √ – Programming: √ – Reading: √	• The configuration clearing command can initialize the setting for prohibition. • The setting for prohibition of block erasure commands is not available. • The setting for prohibition of programming commands is not available. • The setting for prohibition of read commands is not available.	ID authentication is always in effect.
Prohibition of the connection of a dedicated flash memory programmer	• Block erasure commands: × • Programming commands: × • Read commands: ×	• Block erasure: √ • Programming: √ • Reading: √	Since execution of the configuration clearing command is prohibited, initialization of the setting for prohibition is not possible.	Initialization of the settings prohibited is impossible because the configuration clearing command is not supported.
Prohibition of block erasure commands	• Block erasure commands: × • Programming commands: √ • Read commands: √	• Block erasure: √ • Programming: √ • Reading: √	• Since execution of the configuration clearing command is prohibited, initialization of the setting for prohibition is not possible. • The setting for ID authentication to be effective for serial programming is not available.	Initialization of the settings prohibited is impossible because the configuration clearing command is not supported.
Prohibition of programming commands	• Block erasure commands: ×*1 • Programming commands: × • Read commands: √	• Block erasure: √ • Programming: √ • Reading: √	• Executing the configuration clearing command only can initialize the settings prohibited. • The setting for ID authentication to be effective for serial programming is not available.	Initialization of the settings prohibited is impossible because the configuration clearing command is not supported.
Prohibition of read commands	• Block erasure commands: √ • Programming commands: √ • Read commands: ×	• Block erasure: √ • Programming: √ • Reading: √		

Note 1. Block erasure commands can only be executed by erasing all user areas in the following order starting from block 0: erasure of all areas → erasure of user boot area → erasure of data areas starting from block 0. The prohibition setting can only be initialized by issuing the configuration clearing command.

The on-chip flash memory supports various security functions. **Table 33.6, Summary of Protection Functions** lists protection functions that are supported by the on-chip flash memory.

Table 33.6 Summary of Protection Functions

Function	Description
Block protection	Lock bit settings can be individually made to enable or disable programming/erasure of each block of the user area of code flash memory. Programming and erasure by self programming of an area for which the lock bit is set and the lock bit function is enabled are prohibited. Programming or erasure can proceed again when the lock bit function is disabled after having been enabled. When a block of code flash memory is erased, the lock bit for that block is also erased.
User boot protection	Programming or erasure of the user boot area by self programming is prohibited. Programming or erasure of the user boot area by serial programming is available.

33.5 Serial Programming

A dedicated flash memory programmer can be used to program the flash memory in serial programming mode.

Serial Programming

The microcontroller is mounted on the system board at the time of serial programming. Providing a connector to the board enables programming of the target microcontroller by the flash memory programmer to proceed.

33.5.1 Environments for Programming

The recommended environments for programming the flash memory of the microcontroller with data are described below.

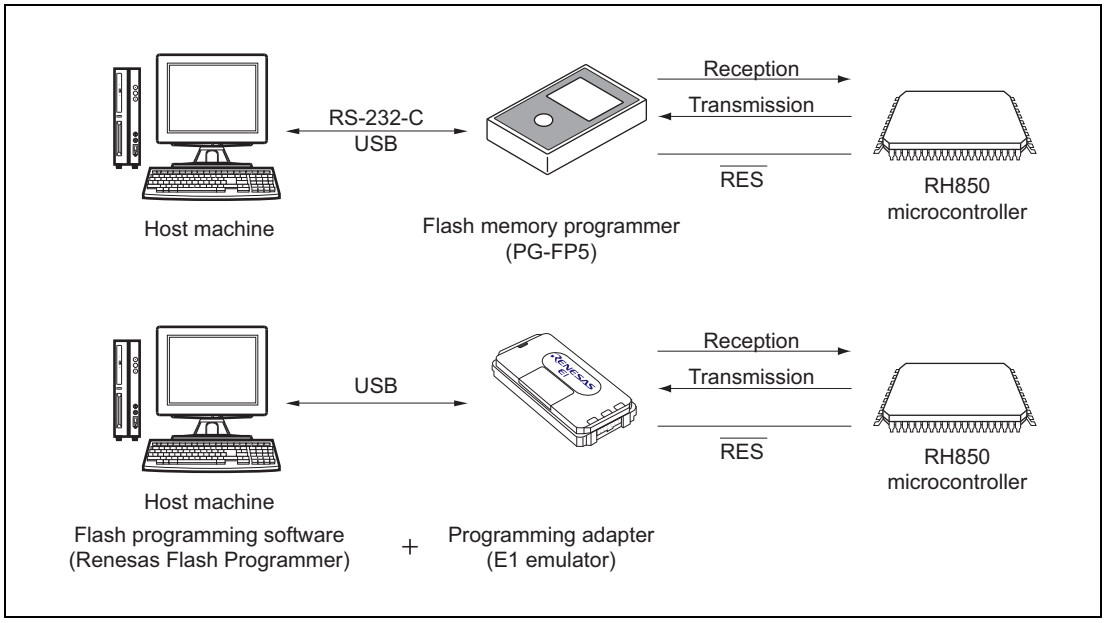


Figure 33.4 Environments for Programming Flash Memory

By using the PG-FP5 flash memory programmer or the combination of the Renesas Flash Programmer (software for writing to flash memory) running on the host machine and the E1 emulator as an programming adapter, the user is easily able to erase, program, and verify the contents of the on-chip memory of flash-memory-equipped microcontrollers from Renesas Electronics.

The PG-FP5 flash memory programmer handles programming from a host machine or programming in stand-alone mode while the Renesas Flash Programmer only handles programming from a host machine.

NOTE

For details on the PG-FP5, see the *PG-FP5 Flash Memory Programmer User's Manual*. For details on the Renesas Flash Programmer of flash programming software, see the *Renesas Flash Programmer Flash Programming Software User's Manual*.

33.6 Selection of the Communication Method

With this product, 2-wire UART connection or clock synchronous connection is selectable as a method of serial communications by the level on the FLMODE pin. For details of the FLMODE pin setting, see **Section 5, Operating Mode**. For setting of programming environment in line with each communication method, see the *PG-FP5 Flash Memory Programmer User's Manual* and *Renesas Flash Programmer Flash Programming Software User's Manual*.

33.7 Self-Programming

33.7.1 Overview

This product supports programming of the flash memory by user program itself. The code flash and data flash memory can be programmed by using the commands of flash memory application command interface (FACI) for flash memory programming in user's applications. Therefore, update of the user program and programming of constant data fields can be possible.

When the data flash memory is programmed, the background operation facility makes it possible to execute a programming program from the code flash memory to program the data flash memory. Furthermore, the programming program can be copied to the local RAM or the global RAM in advance and executed to program the data flash memory.

For comprehensive information on flash self-programming, see the *RH850/E1x Flash Memory User's Manual: Hardware Interface* which this product targets.

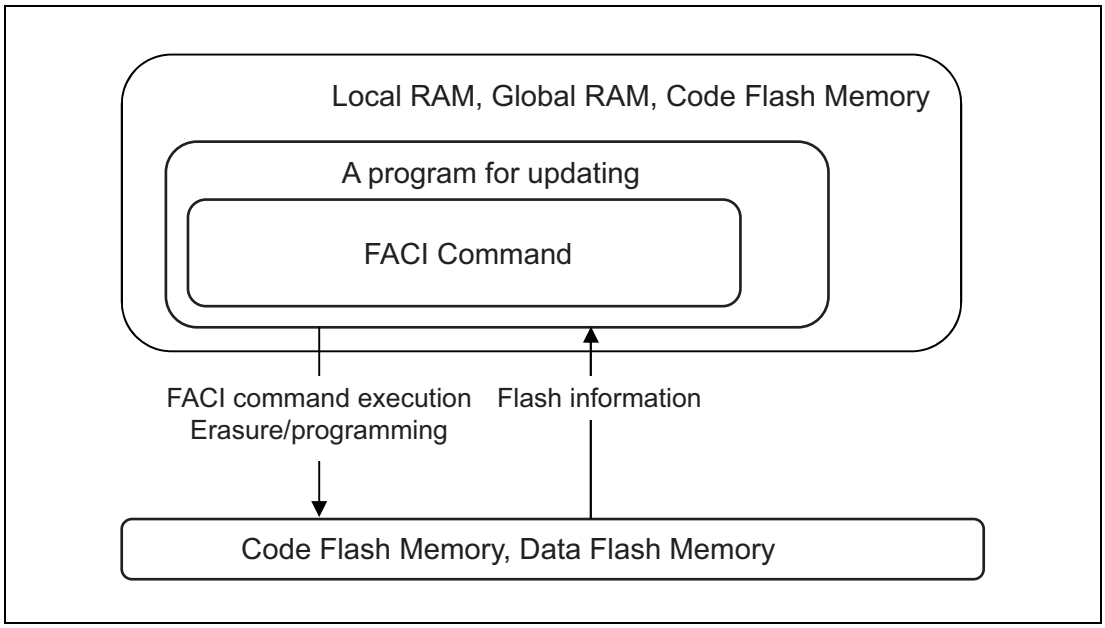


Figure 33.5 Concept of Self-Programming

33.7.2 Background Operation

Background operations can be used when the combination of the flash memory for writing and the flash memory for reading is any of those listed below.

Table 33.7 Conditions under which Background Operation is Usable

Range for Writing	Range for Reading
Data flash memory	Code flash memory

33.8 Reading Flash Memory

33.8.1 Reading Code Flash Memory

Special settings are not required to read code flash memory in normal mode and user boot mode. Data can simply be read out through access to addresses in the code flash memory.

Reading from an area of code flash memory that has been erased but not yet been programmed again (i.e. that is in the non-programmed state) can lead to the detection of an ECC error and generation of the corresponding exception. Furthermore, since the values of data cannot be guaranteed when an ECC error has been generated, use blank checking when you need to confirm that an area is in the non-programmed state. See **Section 29, Safety** for the details on ECC.

33.8.2 Reading Data Flash Memory

Configure the number of read cycles in the FRDCYCLD register prior to reading data from data flash memory in normal mode and user boot mode. Once this register is properly configured, data can be read by simply accessing addresses in the data flash memory. Once the setting for the number of cycles is made, data can simply be read out through access to addresses in the data flash memory.

Values read from data flash memory that has been erased but not yet been programming again are undefined. Use blank checking when you need to confirm that an area is in the non-programmed state.

33.9 Description of Registers

33.9.1 Registers Related to Data Flash Memory

Table 33.8 shows the list of registers related to data flash memory.

Table 33.8 List of Registers Related to Data Flash Memory

Register Name	Abbreviation	R/W	Value after rest	Address	Access Size
Data flash memory read cycle setting register	FRDCYCLD	R/W	$0F_H$	FFC5 9810_H	8

33.9.1.1 FRDCYCLD — Data Flash Memory Read Cycle Setting Register

Access: This register can be read/written in 8-bit units.

Address: FFC5 9810_H

Value after reset: $0F_H$

Bit	7	6	5	4	3	2	1	0
	—	—	—	—	FRDCYCLD[3:0]			
Value after reset	0	0	0	0	1	1	1	1
R/W	R	R	R	R	R/W	R/W	R/W	R/W

Table 33.9 FRDCYCLD Register Contents

Bit Position	Bit Name	Function
7 to 4	—	Received
3 to 0	FRDCYCLD[3:0]	Number of Data Flash Memory Read Cycles Data flash memory is read in setting value + 1 cycles. 00_H to 02_H: Setting prohibited 03_H: Read cycle 4 04_H: Read cycle 5 05_H: Read cycle 6 06_H: Read cycle 7 07_H: Read cycle 8 08_H: Read cycle 9 09_H to $0F_H$: Read cycle 10

33.9.2 Registers Related to Write and Erase Protection of Flash Memory

Table 33.10 shows the list of registers related to write and erase protection of flash memory.

Table 33.10 Registers Related to Write and Erase Protection of Flash Memory

Register Name	Abbreviation	R/W	Value after Reset	Address	Access Size
FHVE15 control register	FHVE15	R/W	0000 0000_H	FFF8 $A430_H$	32
FHVE3 control register	FHVE3	R/W	0000 0000_H	FFF8 2410_H	32

33.9.2.1 FHVE15 — FHVE15 Control Register

FHVE15 is a readable/writable register for software protection of flash memory against programming and erasure. Set both the FHVE15 and FHVE3 registers in the programmable and erasable state ($0000\ 0001_H$) to program or erase flash memory.

Access: This register can be read/written in 32-bit units.

Address: FFF8 $A430_H$

Value after reset: 0000 0000_H

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	FHVE 15CNT
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 33.11 FHVE15 Register Contents

Bit Position	Bit Name	Function
31 to 1	—	Reserved
0	FHVE15CNT	0: Programming and erasure are disabled. 1: Programming and erasure are enabled.

33.9.2.2 FHVE3 — FHVE3 Control Register

FHVE3 is a readable/writable register for software protection of flash memory against programming and erasure. Set both the FHVE15 and FHVE3 registers in the programmable and erasable state ($0000\ 0001_H$) to program or erase flash memory.

Access: This register can be read/written in 32-bit units.

Address: $FFF8\ 2410_H$

Value after reset: $0000\ 0000_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	FHVE3 CNT
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 33.12 FHVE3 Register Contents

Bit Position	Bit Name	Function
31 to 1	—	Reserved
0	FHVE3CNT	0: Programming and erasure are disabled. 1: Programming and erasure are enabled.

33.9.3 Registers Related to Product Information

Table 33.13 shows the list of registers related to product information.

Table 33.13 List of Registers Related to Product Information

Register Name	Abbreviation	R/W	Value after Reset	Address	Access Size
Product name storage register (1)	PRDNAME1	R	See **Table 33.14**	FFCD 00D0$_H$	32
Product name storage register (2)	PRDNAME2	R	See **Table 33.14**	FFCD 00D4$_H$	32
Product name storage register (3)	PRDNAME3	R	See **Table 33.14**	FFCD 00D8$_H$	32
Product name storage register (4)	PRDNAME4	R	See **Table 33.14**	FFCD 00DC$_H$	32

Table 33.14 Relationship between Product Names and PRDNAME Initial Value (Value after Reset)

Product Name	PRDNAME4	PRDNAME3	PRDNAME2	PRDNAME1
R7F701202	2020 2020$_H$	2020 2032$_H$	3032 3130$_H$	3746 3752$_H$
R7F701204	2020 2020$_H$	2020 2034$_H$	3032 3130$_H$	3746 3752$_H$

33.9.3.1 PRDNAMEn; n = 1 to 4 — Product Name Storage Register

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	PRDNAMEn [31:16] (n = 1 to 4)															
Value after reset *1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	PRDNAMEn [15:0] (n = 1 to 4)															
Value after reset *1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1	0/1
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Note 1. This is the value indicated in **Table 33.14, Relationship between Product Names and PRDNAME Initial Value (Value after Reset)**.

Table 33.15 List of Registers Related to Product Information

Bit Position	Bit Name	Function
31 to 0	PRDNAME[31:0]	Product name: Indicate product model names by using 16-byte ASCII code. PRDNAME1[31:0]: Fourth to first bytes of product model name PRDNAME2[31:0]: Eighth to fifth bytes of product model name PRDNAME3[31:0]: Twelfth to ninth bytes of the product model name PRDNAME4[31:0]: Sixteenth to thirteenth bytes of the product model name

33.10 Option Bytes

The flash memory has the extended area (option bytes) to store data specified by the user for various purposes. Changes in settings such as initial setting of peripheral functions using option bytes become effective after release from the reset state. **Table 33.16** shows option byte setting area. Reserved area is always read as 1. When writing, always write 1. For setting and reading option bytes, see the *PG-FP5 Flash Memory Programmer User's Manual* and *Renesas Flash Programmer Flash Programming Software User's Manual*, or the *RH850/E1x Flash Memory User's Manual: Hardware Interface*.

Table 33.16 Setting Area of Option Bytes

Option Byte Area (each 8 bits × 32 = Total 256 bits)	Setting Effective Area	Initial State of Shipped Product*1
Option bytes 4 to 1 (OPBT0)	Enable	7FFF FFFE$_H$
Option bytes 8 to 5 (OPBT1)	Reserved	FFFF FFFF$_H$
Option bytes 12 to 9 (OPBT2)	Enable	FFFF FFFF$_H$
Option bytes 16 to 13 (OPBT3)	Reserved	FFFF FFFF$_H$
Option bytes 20 to 17 (OPBT4)	Reserved	FFFF FFFF$_H$
Option bytes 24 to 21 (OPBT5)	Reserved	FFFF FFFF$_H$
Option bytes 28 to 25 (OPBT6)	Reserved	FFFF FFFF$_H$
Option bytes 32 to 29 (OPBT7)	Reserved	FFFF FFFF$_H$

Note 1. This is the initial value of the shipped product. The value can be modified by setting the option bytes.

33.10.1 OPBT0 — Option Byte 4 to 1 Bit Arrangement

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	OPWD RUN	OPWD INT	OPWD WS1	OPWD WS0	OPWD OVF2	OPWD OVF1	OPWD OVF0	—	—	—	—	—	—	—	—	—
Initial state*1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W*2	R/W*2	R/W*2	R/W*2	R/W*2	R/W*2	R/W*2	R/W*2	R/W*2	R/W*2	R/W*2	R/W*2	R/W*2	R/W*2	R/W*2	R/W*2

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	AUDR EN	—	—	STM SEL1	STM SEL0
Initial state*1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0
R/W	R/W*2	R/W*2	R/W*2	R/W*2	R/W*2	R/W*2	R/W*2	R/W*2	R/W*2	R/W*2	R/W*2	R/W*2	R/W*2	R/W*2	R/W*2	R/W*2

Note 1. This is the initial value of the shipped product. The value can be modified by setting the option bytes.

Note 2. Only when an FACI command is used and when the PG-FP5 flash memory programmer and Renesas Flash Programmer flash programming software are used, reading and setting (R/W) are possible.

Table 33.17 OPBT0 Contents

Bit Position	Bit Name	Function
31	OPWDRUN	This bit selects the start mode of WDTA0. 0: WDTA0 software trigger start mode 1: WDTA0 default start mode
30	OPWDINT	This bit enables or disables a 75% interrupt request of WDTA0 (WDTA0TIT). 0: WDTA0TIT is disabled. 1: WDTA0TIT is enabled.
29, 28	OPWDWS1, OPWDWS0	These bits select the window-open period of WDTA0.

OPWDWS1	OPWDWS0	Window-Open Period
0	0	25%
0	1	50%
1	0	75%
1	1	100%

27 to 25	OPWDOVF2 to OPWDOVF0	These bits select the overflow interval time of WDTA0.

OPWDOVF2	OPWDOVF1	OPWDOVF0	Overflow Interval Time
0	0	0	2^9 / WDTATCKI
0	0	1	2^{10} / WDTATCKI
0	1	0	2^{11} / WDTATCKI
0	1	1	2^{12} / WDTATCKI
1	0	0	2^{13} / WDTATCKI
1	0	1	2^{14} / WDTATCKI
1	1	0	2^{15} / WDTATCKI
1	1	1	2^{16} / WDTATCKI

24 to 5	—	Reserved (The setting value should be 1.)
4	AUDREN	AUDRAM monitor enable 0: AUDRAM monitor is disabled. 1: AUDRAM monitor is enabled.
3, 2	—	Reserved (The setting value should be 1.)
1, 0	STMSEL1, STMSEL0	These bits select operating mode and startup area. When all of the MD0, MD1, and FLMODE pins are 0, the following operating mode and startup area are selected depending on the combination of the STMSEL1 and STMSEL0. For details, see **Section 5, Operating Mode**.

33.10.2 OPBT2 — Option Byte 12 to 9 Bits Arrangement

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	OPJTAG1	OPJTAG0	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial state*1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W*2	R/W*2	R/W*2	R/W*2	R/W*2	R/W*2	R/W*2	R/W*2	R/W*2	R/W*2	R/W*2	R/W*2	R/W*2	R/W*2	R/W*2	R/W*2

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Initial state*1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
R/W	R/W*2	R/W*2	R/W*2	R/W*2	R/W*2	R/W*2	R/W*2	R/W*2	R/W*2	R/W*2	R/W*2	R/W*2	R/W*2	R/W*2	R/W*2	R/W*2

Note 1. This is the initial value of the shipped product. The value can be modified by setting the option bytes.

Note 2. Only when an FACI command is used and when the PG-FP5 flash memory programmer and Renesas Flash Programmer flash programming software are used, reading and setting (R/W) are possible.

Table 33.18 OPBT2 Register Contents

Bit Position	Bit Name	Function
31	—	Reserved (The setting value should be 1.)
30, 29	OPJTAG1, OPJTAG0	Switch the debug interfaces. The following debug interface is selected depending on the combination of the values of OPJTAG1 and OPJTAG0.
28 to 0	—	Reserved (The setting value should be 1.)

OPJTAG1	OPJTAG0	Debug Interface
0	0	FLSCI3 (writer I/F)
0	1	LPD (4-pin)
1	0	Setting prohibited
1	1	Nexus (JTAG)

33.11 Usage Notes

(1) Reading areas where programming or erasure was interrupted

When programming or erasure of an area of flash memory is interrupted, the data stored in the area become undefined. To avoid undefined data that are read out becoming the source of faulty operation, take care not to fetch instructions or read data from areas where programming or erasure was interrupted.

(2) Reading the code flash memory that has been erased but not yet been programming again

Note that reading from an area of code flash memory that has been erased but not yet been programming again (i.e. that is in the non-programmed state) can lead to the detection of an ECC error and generation of the corresponding exception. Use blank checking when you need to confirm that an area is in the non-programmed state.

(3) Prohibition of additional writing

Writing to a given area twice is not possible. If you want to overwrite data in an area of flash memory after writing to the area has been completed, erase the area first.

(4) Resets during programming and erasure

In the case of an internal or external reset during programming and erasure, wait for at least the minimum width of reset pulse once the operating voltage is within the range stipulated in the electrical characteristics before releasing the device from the reset state.

(5) Allocation of vectors for interrupts and other exceptions during programming and erasure

If an interrupt or other exception is generated during transfer of FCU firmware or during programming or erasure of code flash memory, mask the interrupt or other exception beforehand or allocate the interrupt handler address table* and exception handlers to a space from which instructions can be fetched other than code flash memory.

* This applies to cases where the table reference method is used as the method of selection for the addresses of interrupt handlers. For details, see the *RH850G3M User's Manual: Software*.

(6) Abnormal termination of programming and erasure

Even if programming/erasure ends abnormally due to the generation of an external reset or power cutoff, the programming or erasure state of the flash memory with undefined data cannot be verified or checked. For the area where programming or erasure ends abnormally, the blank check function cannot judge whether the area is erased successfully or not. Erase the area again to prove that the corresponding area is completely erased before using.

If programming and erasure of code flash memory are not completed normally, the lock bit for the target area may be enabled (locked). In such cases, erase the block to erase the lock bit while the lock bit is in the disabled state (the area is not locked).

(7) Items prohibited during programming, erasure, and blank checking

Do not perform the following operations during programming, erasure, and blank checking.

- Have the operating voltage from the power supply go beyond the allowed range.
- Update the values of FHVE15 and FHVE3.
- Change the operating frequency of peripheral clock.

(8) Flash memory commands prohibited before completion of clock gear-up sequence

Various commands during serial programming and self-programming must be executed after clock gear-up sequence is complete. For details of clock gear-up sequence, see **Section 11, Clock Controller**.

(9) Securing coherency of instruction cache and data buffer

After the code flash memory is programmed or erased, a reset or instruction cache and data buffer clearing are required to secure coherency of the instruction cache and data buffer. For details of instruction cache and data buffer, see **Section 3, CPU System**.

Section 34 Flash Security

To protect the code flash memory, data flash memory, and ID codes, this product has the functions described in **Section 33, Flash Memory**, and functions to restrict connection of the debug interface, which are described in this section.

For detailed descriptions of the security functions in serial programming mode and how to program the flash memory, see **Section 33, Flash Memory**.

In this section, the interfaces that may be used in on-chip debugging (the Nexus and 4-pin LPD) are referred to as “debug interfaces”. In addition, ID authentication to protect the code flash memory, data flash memory, and ID codes in user boot mode is referred to as “self ID authentication”, and ID authentication to protect against access through the on-chip debug functions is referred to as “OCD ID authentication”.

The data length of each ID code during self ID and OCD ID authentication as well as the ID code in the ID authentication stage of serial programming is 128 bits. In the initial state of the product at shipment, the ID codes are FFFF_FFFF_FFFF_FFFF_FFFF_FFFF_FFFF_FFFF$_H$.

34.1 Features

34.1.1 Protection of Code Flash Memory, Data Flash Memory, and ID Codes

This product includes the following security functions in user boot mode and serial programming mode to prevent any leaking of the user program programmed in the code flash memory.

34.1.1.1 Functions Unique to User Boot Mode

Self ID authentication protects against programming and erasure of code flash memory and reading of ID code.

(1) Security States

This mode has two security states, protection locked and protection unlocked. Transition between these two states can be made by changing self ID authentication and the ID codes.

1. Protection unlocked
   Security functions have been unlocked by self ID authentication so that protection against programming and erasure of code flash memory and reading of ID codes is not applied.
2. Protection locked
   Security functions have been enabled by self ID authentication and protection against programming and erasure of code flash memory and ID code reading is applied.

34.1.1.2 Functions Unique to Serial Programming Mode

Three functions are provided as security functions unique to serial programming mode: ID authentication, prohibition of programming, erasure, and read commands, and prohibition of serial programmer connection. Parallel use of these functions is not allowed.

(1) ID Authentication

ID codes are checked for authenticity to protect the code flash memory and data flash memory. Programming, erasure, and reading of the code flash and data flash memory can proceed upon successful ID authentication.

(2) Prohibition of Programming, Erasure, and Read Commands

Issuing of programming, erasure, and read commands can be enabled or disabled individually for the code flash memory and data flash memory. All commands are prohibited for both areas in the initial state of the product as shipped.

(3) Prohibition of Serial Programmer Connection

Issuing commands for programming, erasure, or reading to the code flash memory and data flash memory can be disabled in serial programming mode. Transitions to other functions are not possible once this state is set.

34.1.1.3 Common Function of User Boot Mode and Serial Programming Mode

(1) OTP (One Time Programming)

Setting an area of the code flash memory as OTP protects it against any further programming and erasure. For details of the setting, see **Section 33, Flash Memory**.

34.1.2 Connection Restriction Function of Debug Interface

This product is provided with unauthorized access protection via debug interfaces and two security levels are available.

- Security level 1:
  Debug interfaces can be used. At this level, the on-chip debugging function is protected by OCD ID authentication. For OCD to be used, it must be unlocked by using OCD ID authentication.
- Security level 2:
  At this level, the debug interfaces cannot be used.

As described above, restrictions on the connection of code flash and data flash memory, ID code protection, and availability of the debug interfaces differ according to the mode. **Table 34.1** summarizes security functions in each mode.

Table 34.1 Security Functions in Each Mode

Operation Mode	Code Flash, Data Flash, and ID Code Protection	Restriction on Debug Interface Connection
User boot mode	• SELF ID authentication • OTP (Parallel use of OTP with other security features is possible.)	• Security level 1 (OCD ID authentication) • Security level 2 (Debug interface connection is prohibited)
Serial programming mode	• ID authentication • Programming commands, block erasure commands, and read commands are prohibited. • Connection of serial programmers is prohibited. (The above three cannot be used in parallel.) • OTP (parallel use possible)	• No function (Debug interface connection is always prohibited.)

34.2 Security in User Boot Mode

34.2.1 SELF ID Authentication

To prevent leakage of the user program written to the code flash memory, this product has security functions to enable or disable the programming and erasure of code flash memory and the reading of ID codes. The setting of enabling or disabling of these functions can be switched through self ID authentication with the ID code the user has set as the expected value.

34.2.2 SELF ID Authentication and Security State

Table 34.2 and **Figure 34.1** show the security state by self ID authentication and conditions for transition.

Table 34.2 Security Setting State

State	Self ID Authentication	Protection against Programming and Erasure of Code Flash and Reading of ID Codes
Protection unlocked	Unlocked	Not protected
Protection locked	Locked	Protected

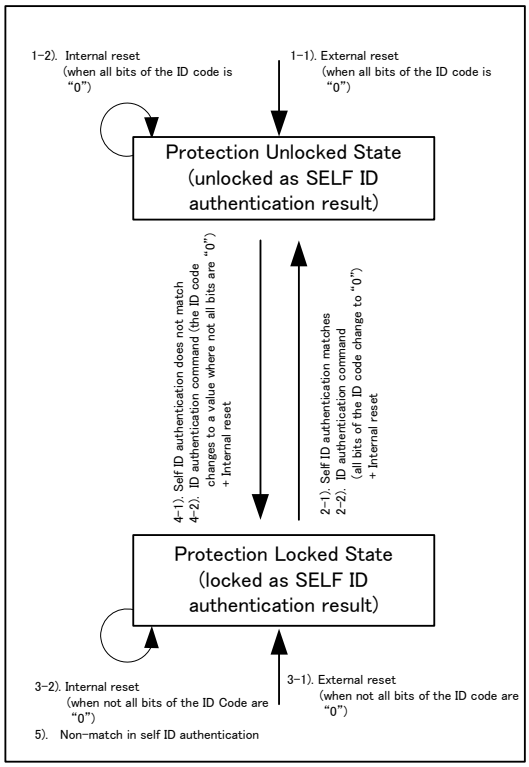


Figure 34.1 Security Setting State by SELF ID Authentication and Transition

The conditions for the transitions of security states shown in **Figure 34.1** are as follows.

- Conditions for Transitions to the Protection Unlocked State

1). Startup when protection is unlocked

1-1). When an external reset is applied while the ID code is all 0s, startup proceeds with protection unlocked.

1-2). When an internal reset is applied without modification of the ID code while all bits of the ID code are 0, startup proceeds with protection in place.

2). Transition from Protection Locked to Protection Unlocked

2-1). A transition to the protection unlocked state follows a match in self ID authentication matches.

2-2). When an internal reset is applied after changing the ID code has been changed from other than “all bits 0” to “all bits 0”, startup proceeds with protection unlocked.

- Conditions for Transitions to Protection Locked

3). Startup with protection in place

3-1). When an external reset is applied while the ID code is other than “all bits 0”, startup proceeds with protection in place.

3-2). When an internal reset is applied without modification of the ID code while not all bits of the ID code are 0, startup proceeds with protection in place.

4). Transition from protection unlocked to protection locked

4-1). A transition to the protection locked state follows a non-match in self ID authentication.

4-2). When an internal reset is applied after the ID code has been changed from other than “all bits 0” to “all bits 0”, startup proceeds with protection in place.

5). To retain protection locked

Protection remains in place in case of a non-match in self ID authentication.

34.3 Security Functions in Serial Programming Mode

For details of security functions in serial programming mode, see **Section 33, Flash Memory**.

34.4 Restricting Connection with Debug Interfaces

This product is capable of restricting connection of the debug interfaces to protect against unauthorized access via the debug interfaces. This feature has two security levels.

- Security level 1: Restrict access to OCD functions by using OCD ID authentication
- Security level 2: Prohibit all connection to the debug interfaces

These security levels can be changed by bits 30 and 29 (OPJTAG1 and OPJTAG0) in option byte 2 in the extended area of the flash memory.
In this section, OPJTAG0 and OPJTAG1 are referred to as the OPJTAG bits.

34.4.1 Security Levels and State of Restricting the Connection of Debug Interfaces

Table 34.3 show each security level and the corresponding security states and **Figure 34.2** shows the conditions for transitions.

Table 34.3 Security Levels and State of Restricting Connection to the Debug Interfaces

State	Result of OCD ID Authentication	OPJTAG Bit *1	Restriction on Debug Interface Connection
Security level 1	Unlocked	Other than 00_B	No restriction
	Locked	Other than 00_B	Restriction on access via the debug interfaces is in place.
Security level 2	—	00_B	Connection to the debug interfaces is prohibited.

Note 1. For a detailed description of the OPJTAG bits, see **Section 33, Flash Memory**.

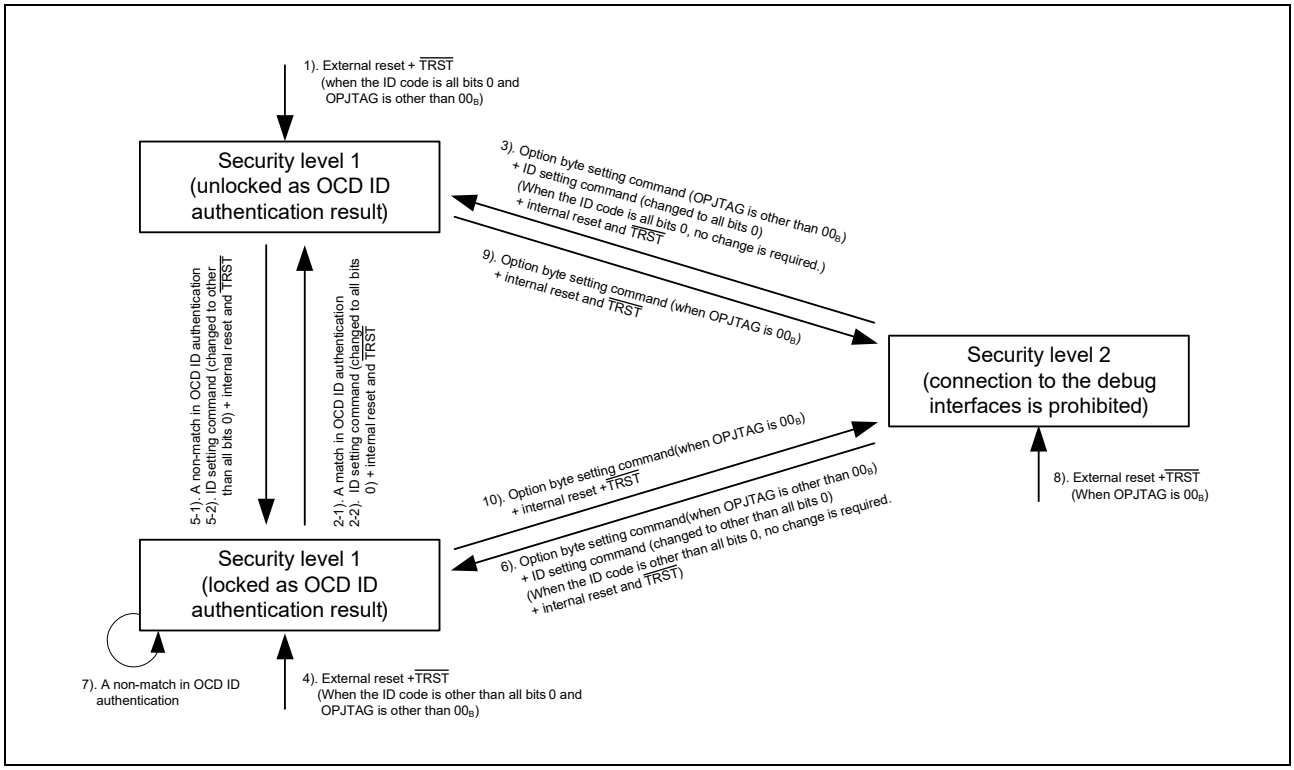


Figure 34.2 Transitions of Security Level

Conditions for transitions to each security level described in **Figure 34.2**.

- Conditions for transitions to security level 1 (locked as OCD ID authentication result)

  1). Startup in the state of security level 1 (unlocked as OCD ID authentication result)
  When an external reset and a reset by $\overline{\text{TRST}}$ are applied while the ID code is all 0s and while the value of the OPJTAG is other than 00_B, startup proceeds with security level 1 (unlocked as OCD ID authentication result).

  2). Transition from security level 1 (locked as OCD ID authentication result) to security level 1 (unlocked as OCD ID authentication result)

  2-1). Transition to security level 1 (unlocked as OCD ID authentication result) follows a match in OCD ID authentication.

2-2). When an internal reset or a reset by $\overline{\text{TRST}}$ is applied after changing the ID code to “all bits 0”, startup proceeds at security level 1 (unlocked as OCD ID authentication result).

3). Transition from security level 2 to security level 1 (unlocked as OCD ID authentication result)
When an internal reset or a reset by $\overline{\text{TRST}}$ is applied after changing the ID code to “all bits 0” and the value of the OPJTAG to other than 00_B, startup proceeds at security level 1 (unlocked as OCD ID authentication result).

- Conditions for Transitions to Security Level 1 (locked as OCD ID Authentication result)

4). Startup in the state of security level 1 (locked as OCD ID authentication result)
When an external reset or a reset by $\overline{\text{TRST}}$ is applied while the ID code is “other than all bits 0” and the value of the OPJTAG is other than 00_B, startup proceeds at security level 1 (locked OCD ID authentication result).

5). Transition from security level 1 (unlocked as OCD ID authentication result) to security level 1 (locked as OCD ID authentication result)

5-1). A transition to security level 1 (locked as OCD ID authentication result) follows a non-match in OCD ID authentication.

5-2). When an internal reset or a reset by $\overline{\text{TRST}}$ is applied after changing the ID code to “other than all bits 0”, startup proceeds at security level 1 (locked as OCD ID authentication result).

6). Transition from security level 2 to security level 1 (locked as OCD ID authentication result)
When an internal reset or a reset by $\overline{\text{TRST}}$ is applied after changing the ID code to “other than all bits 0” and the value of OPJTAG is other than 00_B, startup proceeds at security level 1 (locked as OCD ID authentication result).

7). Retaining security level 1 (locked as OCD ID authentication result)
In case of a non-match in OCD ID authentication, security level 1 (failure in OCD ID authentication) remains in place.

- Transition to security level 2

8). Startup at security level 2
When an external reset or a reset by $\overline{\text{TRST}}$ is applied while the value of OPJTAG is 00 B, startup proceeds at security level 2.

9). Transition from security level 1 (unlocked as OCD ID authentication result) to security level 2
When an internal reset or a reset by $\overline{\text{TRST}}$ is applied after changing the value of the OPJTAG to 00_B, startup proceeds at security level 2.

10). Transition from security level 1 (locked as OCD ID authentication result) to security level 2
When an internal reset or a reset by $\overline{\text{TRST}}$ is applied after changing the value of the OPJTAG to 00_B, startup proceeds at security level 2.

Section 35 RAM

35.1 List of On-Chip RAM

RH850/E1M-S includes the following RAMs.

- Local RAM (PE1): 64 Kbytes
- Local RAM (PCU): 32 Kbytes
- Global RAM: 256 Kbytes (32 Kbytes of data are retained in power-off standby mode)
- Emulation RAM: 8 Kbytes (RAM is not retained)

35.2 Features

Access:

The CPU, PCU, DMAC, and H-Bus can access the local RAM (PE1) and the global RAM.

The PCU also can access the local RAM (PCU).

The access latency from the CPU is the same as that of the local RAM (PE1) and local RAM (self).

For details of address map and access, see **Section 4, Address Space**.

Emulation RAM:

The specific area of the code flash can be replaced with this RAM per page. The access latency from the CPU after replacement is the same as that of the code flash.

The emulation RAM does not include the ECC. Use the emulation RAM only for debugging.

RAM Retention:

32 Kbytes of data in the global RAM are retained in power-off standby mode.

ECC:

The local RAM (PE1), local RAM (PCU), and global RAM include the ECC.

The local RAM (PE1) and global RAM also include the address parity. For details, see **Section 29, Safety**.

35.3 RAM Data Retention

Simultaneous occurrence of write and reset in the RAM space that retains data does not damage RAM data in power-off standby mode.

35.4 Emulation RAM

35.4.1 Emulation RAM

RH850/E1M-S includes the emulation RAM to emulate the code flash. The emulation RAM is placed for each bank of the code flash. RH850/E1M-S has the one-bank code flash and the 8-Kbyte emulation RAM (1 bank). The emulation RAM is available in the code flash emulation function described below, and also can be accessed from CPU1.

35.4.2 Code Flash Emulation Function Using the Emulation RAM

Mapping is enabled and disabled for each bank of the emulation RAM in any area of the code flash using an external debug device and calibration function unit (CFU). Mapping to a code flash area enables a code flash to move to the emulation RAM and it enables the emulation RAM to emulate the code flash. The ROM data can be dynamically modified during execution of a user program via the emulation RAM which has mapped to the code flash area.

Figure 35.1 shows the circuit configuration around the emulation RAM.

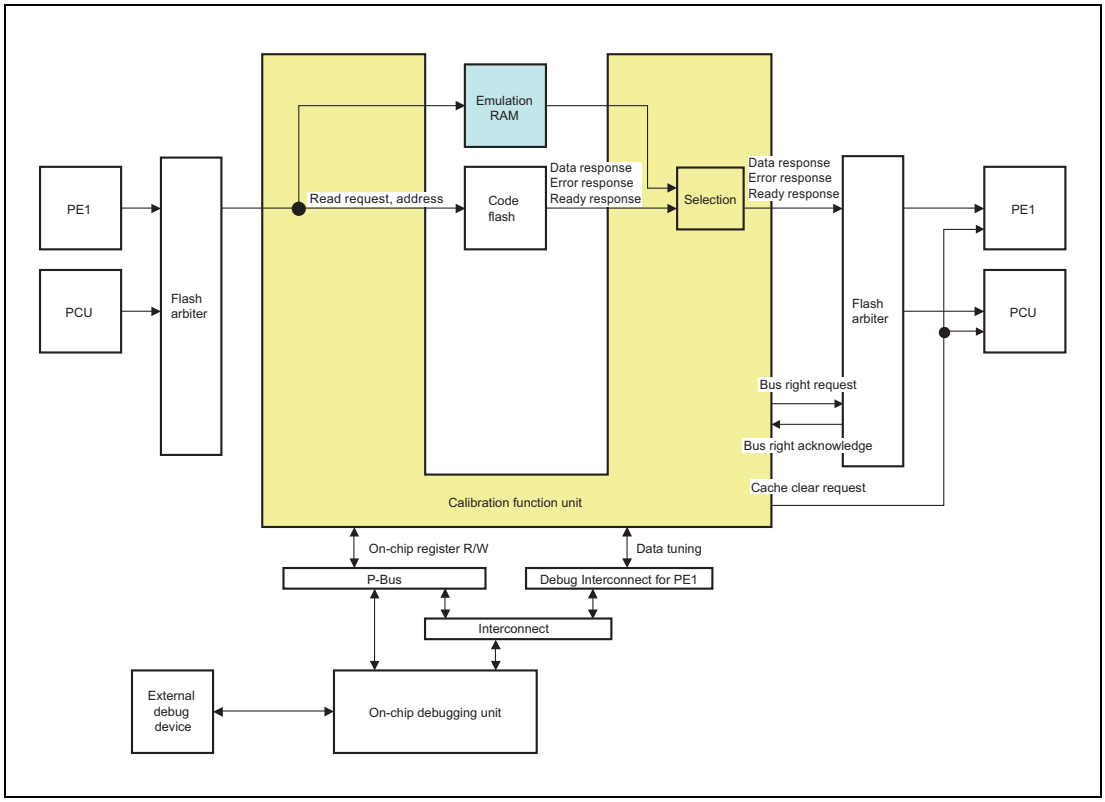


Figure 35.1 Circuit Configuration around the Emulation RAM

35.4.3 Memory Map of Emulation RAM

The memory map of the ERAM in this product is shown below.
The ERAM has 32 banks. Bank 0 only has 8 Kbytes. The ERAM area can be read and written only in 32-bit units.

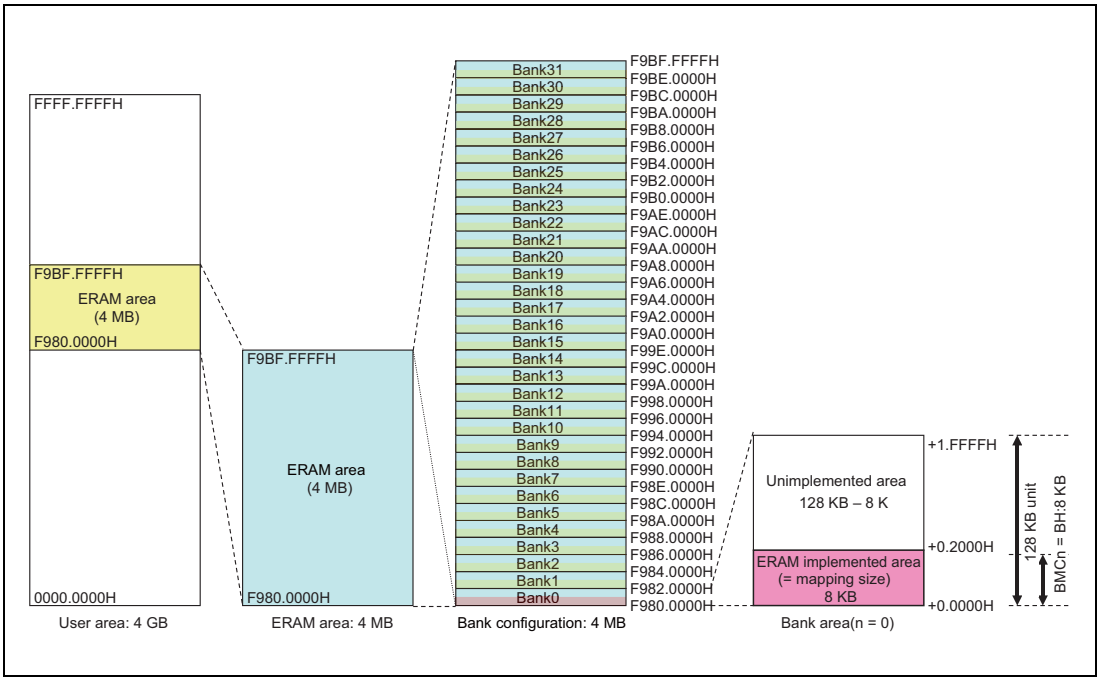


Figure 35.2 Memory Map of 8-KB Emulation RAM

35.4.4 List of CFU Registers

The list below shows function registers that control operations of the calibration function unit (CFU).

The base address of the CFU is FFFF $7800_{H.}$

Table 35.1 List of Function Registers

Address	Register Name	Function
<CFU_base>+08_H	TM_CC	Cache clear Operation register
<CFU_base>+10_H	TM_ME	ERAM bank 0 mapping enable register
<CFU_base>+14_H	TM_MS	ERAM bank 0 mapping status register
<CFU_base>+30_H	TM_BMC0	ERAM bank 0 mapping size setting register
<CFU_base>+40_H	TM_MA0	ERAM bank 0 mapping address register

35.4.5 TM_CC — Cache Clear Operation Register

The TM_CC register is used to issue requests for clearing of the unit (cache or buffer) that holds data from the flash ROM.

Access: This register can be read/written in 32-bit units.

Address: FFFF 7808$_H$

Value after reset 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	CCLR
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 35.2 TM_CC Register Contents

Bit Position	Bit name	Functions
31 to 1	—	Reserved These bits are always read as 0. The write value should always be 0.
0	CCLR	A cache clearing request is issued by writing 1 to this bit. Writing 0 has no effect. The value read is 1 if this bit is read before the response to a cache clearing request is completed and 0 after the response to a cache clearing request is completed. **CAUTION** • After 1 has been written to this bit, writing 1 to the bit is ignored if this is done before the response to the earlier cache clearing request is completed (the cache is not cleared in response to the latter request). • After 1 has been written to this bit, if 0 is written before the response to the cache clearing request is completed, this will not cancel the response to the earlier request 0: A cache clearing request is not being processed or the response to a cache clearing request has been completed (when read). 1: A cache clearing request is issued (when written), or a cache clearing request is being processed (when read).

35.4.6 TM_ME — Tuning Memory Mapping Enable Register

This register is used to control mapping of ERAM bank 0 to the flash memory area.

Access: This register can be read/written in 32-bit units.

Address: FFFF 7810$_H$

Value after reset 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	TME0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R/W

Table 35.3 TM_ME Register Contents

Bit Position	Bit name	Functions
0	TME0	Tuning memory bank n mapping enable (n = 0) These bits control the mapping of ERAM banks n (n = 0) to the user area (flash area). 0: The bank is not mapped. 1: The bank is mapped.

Note 1. When the mapping enable bit TMEn is updated (including writing of its current value again), the cache is cleared.

Note 2. Before ERAM bank n and the tuning memory mapping address register n (TM_Man register) is updated, the corresponding mapping enable bit TMEn must be set to 0 (bank n is not mapped).

Note 3. After an update of the TM_ME register, wait and verify that the value in the TM_MS register has become the same as the setting of the TM_ME register.

35.4.7 TM_MS — Tuning Memory Mapping Status Register

This register is used to indicate the state of mapping to the flash area of ERAM bank 0.

Access: This register can be read in 32-bit units.

Address: FFFF 7814$_H$

Value after reset 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	MES0
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 35.4 TM_MS Register Contents

Bit Position	Bit name	Functions
0	MES0	Tuning memory bank n mapping enable status (n = 0) These bits indicate the state of mapping of ERAM banks n (n = 0) to the user area (flash area). 0: The bank is not being mapped. 1: The bank is being mapped.

35.4.8 TM_BMC0 — Tuning Memory Bank Mapping Size Configuration Register 0

These registers are used to set mapping size of bank 0 of the ERAM.

Access: This register can be read/written in 32-bit units.

Address: FFFF 7830$_H$

Value after reset 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—	—
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R	R

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	—	—	—	—	—	—	—	—	—	—	—	—	BMC0			
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W

Table 35.5 TM_BMC0 Register Contents

Bit Position	Bit name	Functions
3 to 0	BMC0	Bank 0 mapping size setting

Mapping size each bank of the ERAM (banks 0) is set.

0$_H$	Reserved (setting prohibited)	8$_H$	Reserved (setting prohibited)
1$_H$	Reserved (setting prohibited)	9$_H$	8 KB/bank
2$_H$	Reserved (setting prohibited)	A$_H$	Reserved (setting prohibited)
3$_H$	Reserved (setting prohibited)	B$_H$	Reserved (setting prohibited)
4$_H$	Reserved (setting prohibited)	C$_H$	Reserved (setting prohibited)
5$_H$	Reserved (setting prohibited)	D$_H$	Reserved (setting prohibited)
6$_H$	Reserved (setting prohibited)	E$_H$	Reserved (setting prohibited)
7$_H$	Reserved (setting prohibited)	F$_H$	Reserved (setting prohibited)

Only the following combinations of settings for mapping size are available.

ERAM Capacity	Bank 0 Mapping Size Setting
8 Kbytes	9$_H$ (8 Kbytes/bank)
Any combination other than the above	

Note: The corresponding mapping enable bit TMEn must be set to 0 (bank n is not mapped) before updating the tuning memory bank mapping size configuration register (TM_BMC0 register) (n = 0).

35.4.9 TM_MA0 — Tuning Memory Mapping Address Register 0

These registers are used to set mapping addresses for bank 0 of the ERAM.

Access: This register can be read/written in 32-bit units.

Address: FFFF 7840$_H$

Value after reset 0000 0000$_H$

Bit	31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
	TMAn[31:16]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R	R	R	R	R	R	R	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Bit	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
	TMAn[15:0]															
Value after reset	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
R/W	R/W	R/W	R/W	R	R	R	R	R	R	R	R	R	R	R	R	R

Table 35.6 TM_MA0 Register Contents

Bit Position	Bit name	Functions
31 to 0	TMAn[31:0]	Tuning memory mapping address n (n = 0) Specify addresses for mapping ERAM banks n. **CAUTION** TMAn31 to 25 and TMAn12 to 0 are fixed to 0.

BMCn	Mapping Unit	Mapping Judge Bit	Bit to be Ignored
0_H to 8_H	Reserved (setting prohibited)	—	—
9_H	8 Kbytes/bank	TMAn24 to 13	TMAn31 to 25, 12 to 0
A_H to F_H	Reserved (setting prohibited)	—	—

CAUTION

1. **The corresponding mapping enable bit TMEn must be set to 0 (bank n is not mapped) before updating the tuning memory mapping address register n (TM_MAn registers).**
2. **Specify the address in the code flash area as the mapping address. Correct operation is not guaranteed if any other address is specified.**

35.4.10 Flow of Processing for Tuning

The flow of processing for tuning is shown below.

(1)	Mapping initial setting handling
(1-0)	Set the size and address of bank n to be tuned*1 Tuning memory bank mapping size configuration register 0 (TM_BMC0) Tuning memory mapping address register 0 (TM_MA0)
(2)	Handling tuning data
(2-1)	Write data to the ERAM area for use in tuning.
(3)	Mapping the banks
(3-1)	Set the mapping enable bit TME0 in bank 0 for which tuning is complete (n = 0) to 1 (mapping bank 0).
(3-2)	Read the mapping status bit MES0 to verify the completion of mapping settings.

Note 1. Set the corresponding mapping enable bit TME0 to 0 (de-mapping bank 0) to update the register.

When data are tuned (programming of data in the ERAM area) in a mapped bank, the cache must be cleared after tuning of the data.

35.4.11 Notes on Access to ERAM

When accessing from CPU1 to the ERAM area, an LSDC error (lockstep compare error) may occur. The following are the means to mask the error.

(1) Set the $\overline{\text{AUDRST}}$ to H (when the AUD RAM monitor is usable).

(2) Disable interrupts, resets, and error output due to the LSDC error by setting the ECM. Register access by the CFU does not cause the LSDC error.

35.5 Usage Notes

- When ECC error detection/correction is enabled for the local RAM and global RAM, initialize the RAM using a write instruction with the maximum bit length of its access size before using the RAM. For the maximum bit length of the RAM access size, see the applicable data width in **Table 29.1, ECC Overview**.
  If the RAM is accessed before its initialization, an ECC error may be detected. Also if initialization with the maximum bit length is not performed (e.g. if a 32-bit RAM is initialized by an 8-bit or 16-bit access), an ECC error may be detected.
- Buffers are included for high-speed access between the following RAMs and the CPU.
  - Local RAM (PE1), Local RAM (PCU)

  After a value has been written to an address, reading the value from the same address may lead to reading from the buffer instead of the RAM. Ways to make sure that values are being read from the RAM are as follows.
  - For Local RAM (PE1)
    1. Only read the first byte to have been written after having written more than 32 bytes of data.
    2. Issue a SYNCM instruction between a write instruction and an instruction to read from the same address.
  - For Local RAM (PCU)
    1. Only read the first byte to have been written after having written more than 4 bytes of data.
    2. Issue at least one instruction which is not accompanied by access to memory (an NOP instruction, SYNCM instruction, etc.) between a write instruction and an instruction to read from the same address.

Section 36 Boundary Scan

This LSI has the JTAG interface and provides the boundary scan function conforming to the IEEE1149.1 standard.

36.1 Features

- Five test signals (TCK, TDI, TDO, TMS, and $\overline{\text{TRST}}$)
- TAP controller
- Instruction register
- Bypass register
- Boundary scan register

The JTAG interface has six commands.

- BYPASS mode
  Test mode conforming to the IEEE 1149.1
- EXTEST mode
  Test mode conforming to the IEEE 1149.1
- SAMPLE/PRELOAD mode
  Test mode conforming to the IEEE 1149.1
- CLAMP mode
  Test mode conforming to the IEEE 1149.1
- HIGHZ mode
  Test mode conforming to the IEEE 1149.1
- IDCODE mode
  Test mode conforming to the IEEE 1149.1

Figure 36.1 shows a block diagram of the JTAG interface.

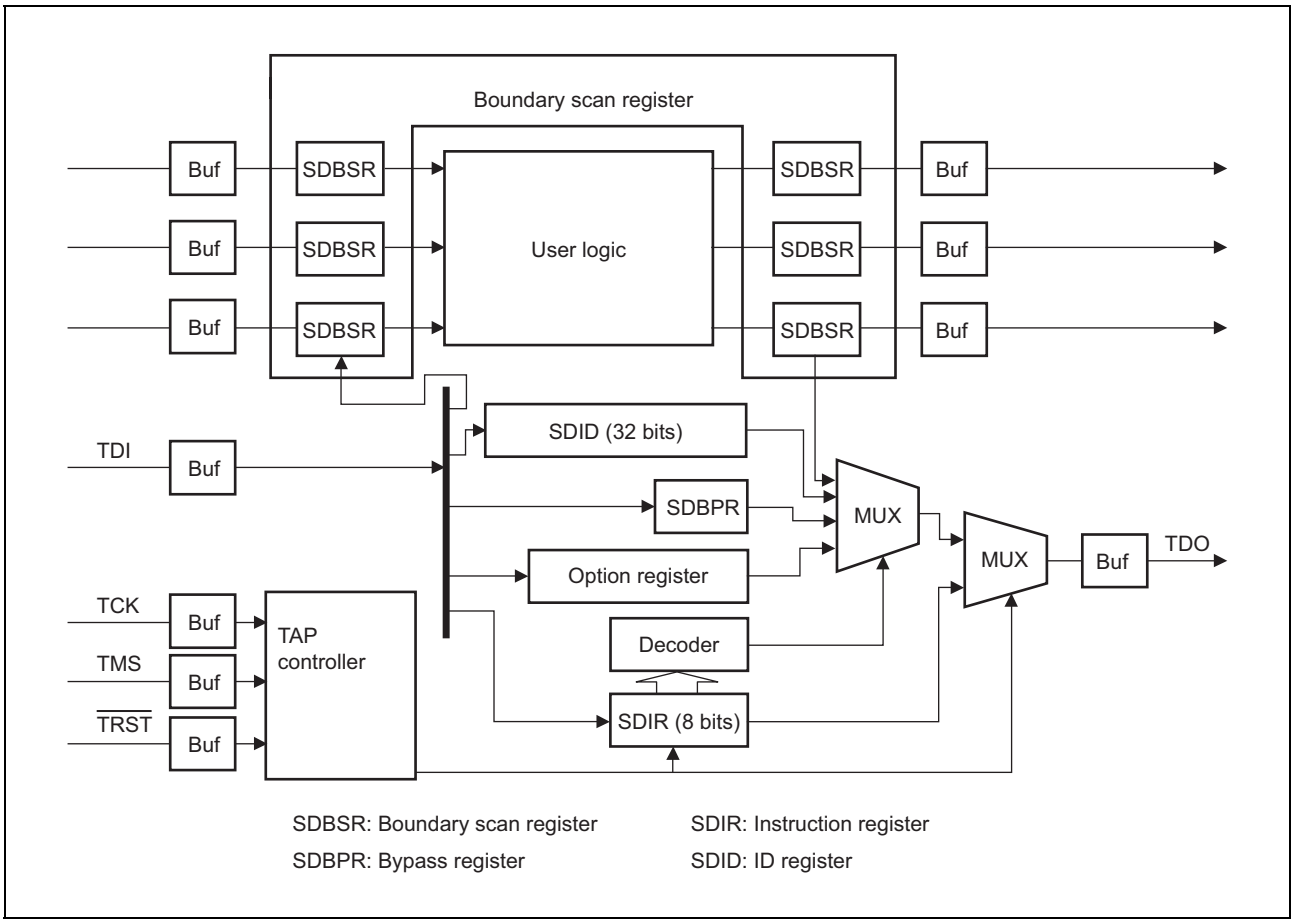


Figure 36.1 Block Diagram of JTAG Interface

36.2 Input/Output Pins

There are five JTAG control signals: TCK, TDI, TMS, TDO, and $\overline{\text{TRST}}$.

Table 36.1 shows the pin configuration.

Table 36.1 Pin Configuration

Pin Name	Description
TCK	Serial data input/output clock pin Data is supplied via the data input pin (TDI) and is output from the data output pin (TDO) in synchronization with this clock signal.
TMS	Mode select input pin Changing the level of this signal in synchronization with TCK changes the state of the TAP controller. For the protocol, refer to **Figure 36.2 TAP Controller State Transition Diagram**.
$\overline{\text{TRST}}$	Reset input pin A low-level input of this signal, which is accepted asynchronously with TCK, resets the JTAG interface. Even if the JTAG interface is not used, $\overline{\text{TRST}}$ must be held low for the specified time at a power on.
TDI	Serial data input pin Changing the state of this pin in synchronization with TCK allows data to be sent to the JTAG interface.
TDO	Serial data output pin Reading the state of this pin in synchronization with TCK allows data to be read from the JTAG interface.

36.3 Register Descriptions

The JTAG interface has the following registers. None of the registers can be accessed by the CPU.

- SDIR: Instruction register
- SDID: ID register
- SDBPR: Bypass register
- SDBSR: Boundary scan register

Table 36.2 Register Configuration

Register Name	Symbol	Access Size	Value after Reset*1
Instruction register	SDIR	8	55_H
ID register	SDID	32	*2
Bypass register	SDBPR	1	Undefined
Boundary scan register	SDBSR	—	Undefined

Note 1. Registers are initialized when the $\overline{\text{TRST}}$ pin is 0 or when TAP is in the test-logic-reset state.
Note 2. It is dependent on products. Contact our sales representative.

Commands can be serially transferred from the serial data input pin (TDI) and input to the instruction register (SDIR). The bypass register (SDBPR) is a 1-bit register, to which TDI and TDO are connected in BYPASS mode, CLAMP mode, and HIGHZ mode. The boundary scan register (SDBSR) is connected to TDI and TDO in SAMPLE/PRELOAD mode and EXTEST mode. The ID code register (SDID) is a 32-bit register, from which the ID code is output via TDO in IDCODE mode.

Table 36.3 shows the serial transfer types possible with the JTAG interface registers.

Table 36.3 Serial Transfer Types Possible with JTAG Interface Registers

Register	Serial Input	Serial Output
SDIR	Possible	Impossible*1
SDBPR	Possible	Possible
SDBSR	Possible	Possible
SDID	Impossible	Possible

Note 1. A fixed value is read out.

36.3.1 SDIR — Instruction Register

SDIR is an 8-bit register that holds a boundary scan command. SDIR is initialized when $\overline{\text{TRST}}$ is asserted or TAP is in the test-logic-reset state. Operation is not guaranteed when any reserved command is set in this register.

Table 36.4 Boundary Scan Commands

IR Code								Description
0	0	0	0	0	0	0	0	JTAG EXTEST
0	1	0	0	0	0	0	0	JTAG SAMPLE/PRELOAD
1	1	0	1	0	0	0	0	JTAG CLAMP
1	0	0	0	0	0	0	0	JTAG HIGHZ
0	1	0	1	0	1	0	1	JTAG IDCODE (value after reset)
1	1	1	1	1	1	1	1	JTAG BYPASS
Other than above								Reserved

36.3.2 SDID — ID Register

SDID is a 32-bit register that indicates an LSI-specific ID.

SDID can be read via the JTAG interface pin when the IDCODE command is set but cannot be written to.

For the read values, refer to **Table 36.2, Register Configuration**.

36.3.3 SDBPR — Bypass Register

SDBPR is a 1-bit register. When SDIR is set to BYPASS mode, SDBPR is connected between TDI and TDO. The value after reset is undefined. SDBPR is not initialized by a reset or asserted $\overline{\text{TRST}}$.

36.3.4 SDBSR — Boundary Scan Register

SDBSR is a shift register located on the PADs for controlling the external I/O pins. When SDIR is set to SAMPLE/PRELOAD or EXTEST mode, SDBSR is connected between TDI and TDO. The value after reset is undefined. SDBSR is not initialized by a reset or asserted $\overline{\text{TRST}}$.

36.4 Operation

36.4.1 TAP Controller

Figure 36.2 shows the TAP controller internal states.

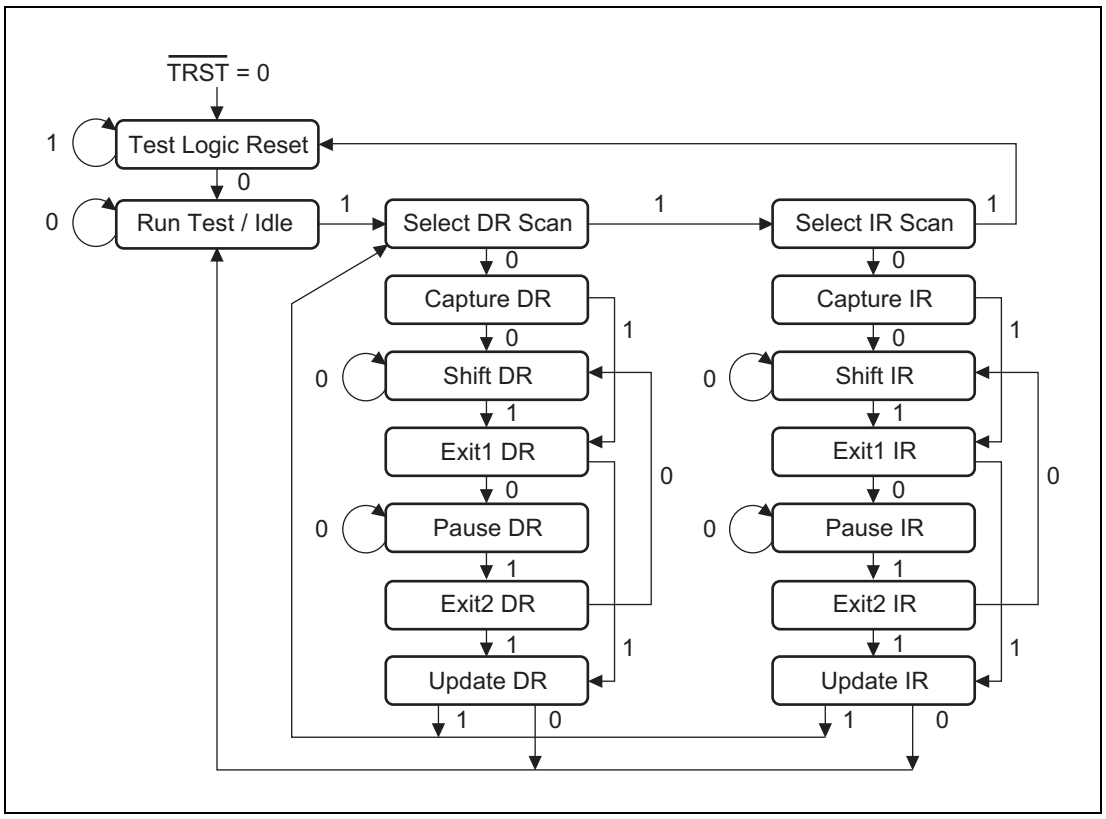


Figure 36.2 TAP Controller State Transition Diagram

Note 1. Transition is made according to the TMS value at the rising edge of TCK. The TDI value is sampled at the rising edge of TCK and is shifted at the falling edge. TDO is in the high-impedance state in the states other than Shift-DR and Shift-IR. Asserting $\overline{\text{TRST}}$ causes transition to the test-logic-reset state asynchronously with TCK.

36.4.2 Supported Commands

36.4.2.1 BYPASS

The BYPASS command is a standard command indispensable to bypass register operation. This command shortens the shift path to achieve high-speed serial data transfer of other LSIs on the printed-circuit board. During execution of this command, the test circuit has no effect on the system circuit.

36.4.2.2 SAMPLE/PRELOAD

The SAMPLE/PRELOAD command is used to input the value to the boundary scan register from the internal circuits of this LSI; to output the value from the scan path; and to load data onto the scan path. During execution of this command, the level of the input pin of this LSI is sent to the internal circuits as is, and the value of the internal circuits is output to the outside via the output pin as is. Executing this command has no effect on the system circuit of this LSI.

The SAMPLE operation allows taking in the snapshots of the value to be transferred to the internal circuits from the input pin or the value to be transferred to the output pin from the internal circuits to the boundary scan register and allows reading the snapshots from the scan path. Snapshots can be taken in without preventing the normal operation of this LSI.

The PRELOAD operation allows setting the initial value to the parallel output latch of the boundary scan register from the scan path prior to the EXTEST command. If the EXTEST command is executed without PRELOAD operation, the undefined value is output from the output pin until the first scan sequence is completed (transfer to the output latch) because the parallel output latch value is always output to the output pin with the EXTEST command.

36.4.2.3 EXTEST

The EXTEST command is used to test the external circuits when this LSI is mounted on the printed-circuit board. When this command is executed, the output pin is used to output the test data (previously set with the SAMPLE/PRELOAD command) from the boundary scan register to the printed-circuit board; whereas the input pin is used to take in the test result from the printed-circuit board to the boundary scan register. When the EXTEST command is executed N times for testing, the test data for the Nth execution is scanned in at the (N – 1)th scan-out.

If the data is loaded onto the boundary scan register of the output pin in the Capture-DR state of this command, it is not used for testing the external circuits (replaced through shift operation).

36.4.2.4 CLAMP

When the CLAMP command is selected, the output pin outputs the boundary scan register value that has been previously set with the SAMPLE/PRELOAD command. While the CLAMP command is selected, the boundary scan register retains the previous state regardless of the TAP controller state.

The bypass register is connected between TDI and TDO and operates in the same manner as when the BYPASS command is selected.

36.4.2.5 HIGHZ

When the HIGHZ command is selected, all the output pins go to the high-impedance state. While the HIGHZ command is selected, the boundary scan register retains the previous state regardless of the TAP controller state.

The bypass register is connected between TDI and TDO and operates in the same manner as when the BYPASS command is selected.

36.4.2.6 IDCODE

The IDCODE command sets the JTAG interface pins to IDCODE mode, which is defined by the JTAG standard. When the JTAG interface is initialized (by asserting $\overline{\text{TRST}}$ or placing TAP in the test-logic-reset state), IDCODE mode is set.

36.4.3 Notes

There are restrictions related to the JTAG interface as follows.

- The power supply/GND pins are not subjected to boundary scan.
- The reference voltage pins of the A/D converter (ADSVREFH, ADSVREFL, A0VREFH, and A1VREFH) are not subjected to boundary scan.
- The pins to be connected to a stabilizing capacitor (ADSVCL and RAMVCL) are not subjected to boundary scan.
- The EPT control pin (EPTVOUT) is not subjected to boundary scan.
- The NC pins are not subjected to boundary scan.

Table 36.5 lists the pins not subjected to boundary scan.

Table 36.5 Pins not Subjected to Boundary Scan

Type	Pins
Analog input	AN000/DSAN0P, AN001/DSAN0N, AN002/DSAN1P, AN003/DSAN1N, AN010/DSAN2P/P14_12, AN011/DSAN2N/P14_13, AN012/DSAN3P/P14_14, AN013/DSAN3N/P14_15, AN020/DSAN4P/P14_0, AN021/DSAN4N/P14_1, AN022/DSAN5P/P14_2, AN023/DSAN5N/P14_3, AN030/DSAN6P/P14_4, AN031/DSAN6N/P14_5, AN032/DSAN7P/P14_6, AN033/DSAN7N/P14_7, AN040/P14_8, AN041/P14_9, AN042/P14_10, AN043/P14_11, AN050/P15_0, AN051/P15_1, AN052/P15_2, AN053/P15_3, AN100, AN101, AN102, AN110, AN111, AN112, AN120, AN121, AN122, AN130, AN131, AN132, AN140, AN141, AN142, AN150/P16_0, AN151/P16_1, AN152/P16_6, AN160/P16_2, AN161/P16_3, AN162/P16_7, AN170/P16_4, AN171/P16_5, AN172/P16_8
Debug system	AUDCK, $\overline{\text{DRDY}}$/LPDCLKO, $\overline{\text{EVTI}}$, $\overline{\text{EVTO}}$, $\overline{\text{RES}}$, TCK/LPDCLKI, TDI/LPDI, TDO/LPDO, TMS/$\overline{\text{EVTO}}$, $\overline{\text{TRST}}$/LPDRST
LVDS	RHSB0FCLN, RHSB0FCLP, RHSB0SON, RHSB0SOP, RHSB1FCLN, RHSB1FCLP, RHSB1SON, RHSB1SOP
Mode setting	FLMODE, MD0, MD1
Clock	CK, EXTAL, XTAL

- The HIGHZ command is invalid for the pulled-down pins.

36.5 Usage Notes

1. Once a command is set, it is not modified until another command is issued again. To continuously issue the same commands, insert a command that has no effect on chip operation (such as BYPASS mode) between the desired commands.
2. To start the system in boundary scan mode, negate $\overline{\text{TRST}}$ while $\overline{\text{RES}}$ is high.
3. For the maximum clock frequency that can be input to TCK, refer to Section 37, Electrical Characteristics.
4. If the number of serially transferred bits exceeds the number of bits of the register connected between TDI and TDO, the data that is input from TDI is output from TDO after the serial data equal to the number of register bits are output.
5. If the serial transfer sequence is corrupted, be sure to reset $\overline{\text{TRST}}$. Here, start the transfer over again regardless of the point of transfer corruption.
6. Data is output via TDO in synchronization with the falling edge of TCK.
7. To facilitate debugging, route $\overline{\text{TRST}}$ on the board in such a way that patterns can be easily cut.
8. No commands can be accepted in power-off standby mode.

Section 37 Electrical Characteristics

37.1 Absolute Maximum Ratings

Table 37.1 lists absolute maximum ratings.

Table 37.1 Absolute Maximum Ratings

Item		Symbol	Rated Value	Unit	Note
Power voltage*1	PLLVCC, SYSVCC, VCC	VCC	–0.3 to +4.3	V	
	EVCC	EVCC	–0.3 to +6.5	V	
	VDD	VDD	–0.3 to +1.8	V	
	LVDVCC	VLVDVCC	–0.3 to +4.3	V	
	TTLVCC	VTTLVCC	–0.3 to +6.5	V	
Input voltage	SYSVCC power pin	Vin	–0.3 to SYSVCC +0.3	V	See **Table 37.3**
	VCC power pin	Vin	–0.3 to VCC +0.3	V	
	EVCC power pins	Vin	–0.3 to EVCC +0.3	V	
	5 V tolerant pin*2	Vin	–0.3 to +5.8	V	
RHSB power voltage		LVDVCC	–0.3 to +4.3	V	
		TTLVCC	–0.3 to +6.5	V	
RHSB input voltage		V_{LVDSIN}	–0.3 to LVDVCC +0.3	V	
		V_{TTLIN}	–0.3 to TTLVCC +0.3	V	
Analog power voltage		A0VCC, A1VCC	–0.3 to +6.5	V	
		ADSVCC	–0.3 to +6.5	V	
Analog reference voltage		A0VREFH	–0.3 to A0VCC +0.3	V	
		A1VREFH	–0.3 to A1VCC +0.3	V	
		ADSVREFH	–0.3 to ADSVCC +0.3	V	
		ADSVREFL	–0.3 to ADSVSS +0.3	V	
Analog input voltage		V_{AIN}	–0.3 to A0VCC +0.3 –0.3 to A1VCC +0.3	V	
		V_{ADSIN}	–0.3 to ADSVCC +0.3	V	
VSS differential voltage*3 (Condition: Between any two of VSS, A0VSS, A1VSS, ADSVSS, PLLVSS, and LVDVSS)			–0.1 to +0.1	V	
Maximum input current (per pin)	Digital input pin	Imax	–25 to +25	mA	Only 1 pin simultaneously
	Analog input pin	Imax	–25 to +25	mA	
Junction temperature*1		Tj	–40 to +150	°C	
Storage temperature		Tstg	–55 to +150	°C	After mounting

Note 1. Cumulative hours of operation of this LSI with Tj in the range from 125°C to 150°C must be kept within 3000 hours.

Note 2. Pins below described as “(5 V tol.)” in **Table 37.2, Relationship between Power Name and Pin**.
FLMODE, MD0, MD1, $\overline{RES}$, $\overline{TRST}$, NMI

Note 3. Of the digital power supplies on the board, short-circuit PLLVSS to VSS.

NOTE

Using this LSI without observing these absolute maximum ratings may result in permanent breakdown of the LSI. This product is used in combination of multiple power voltages simultaneously in some cases. Use this LSI conforming to power pin connections, conditions for combination of power voltages to be applied, voltages that can be applied to pins, and output voltage conditions, which are specified in the manual. Using this LSI with unspecified power connection or voltage may result in permanent breakdown of the LSI or damage to the system that contains this LSI.

37.2 DC Characteristics

37.2.1 Relationship between Power Name and Pin

Table 37.2 shows the relationship between power name and pin.

Table 37.2 Relationship between Power Name and Pin (1/7)

Pin Name (Value after Reset)	Port Group	Circuit Power Name	I/O	Maximum Input Voltage (V)	Input Buffer Type	Output Driving Ability	Note
AN110	—	A1VCC	I	A1VCC+0.3	SAR analog		
AN102	—	A1VCC	I	A1VCC+0.3	SAR analog		
AN101	—	A1VCC	I	A1VCC+0.3	SAR analog		
AN100	—	A1VCC	I	A1VCC+0.3	SAR analog		
AN053/P15_3	P15	A0VCC	I	A0VCC+0.3	SAR analog SchB		
AN052/P15_2	P15	A0VCC	I	A0VCC+0.3	SAR analog SchB		
AN051/P15_1	P15	A0VCC	I	A0VCC+0.3	SAR analog SchB		
AN050/P15_0	P15	A0VCC	I	A0VCC+0.3	SAR analog SchB		
AN043/P14_11	P14	A0VCC	I	A0VCC+0.3	SAR analog SchB		
AN042/P14_10	P14	A0VCC	I	A0VCC+0.3	SAR analog SchB		
AN041/P14_9	P14	A0VCC	I	A0VCC+0.3	SAR analog SchB		
AN040/P14_8	P14	A0VCC	I	A0VCC+0.3	SAR analog SchB		
AN033/DSAN7N/ P14_7	P14	ADSVCC	I	ADSVCC+0.3	ΔΣ analog SchB		
AN032/DSAN7P/ P14_6	P14	ADSVCC	I	ADSVCC+0.3	ΔΣ analog SchB		
AN031/DSAN6N/ P14_5	P14	ADSVCC	I	ADSVCC+0.3	ΔΣ analog SchB		
AN030/DSAN6P/ P14_4	P14	ADSVCC	I	ADSVCC+0.3	ΔΣ analog SchB		
AN023/DSAN5N/ P14_3	P14	ADSVCC	I	ADSVCC+0.3	ΔΣ analog SchB		
AN022/DSAN5P/ P14_2	P14	ADSVCC	I	ADSVCC+0.3	ΔΣ analog SchB		
AN021/DSAN4N/ P14_1	P14	ADSVCC	I	ADSVCC+0.3	ΔΣ analog SchB		
AN020/DSAN4P/ P14_0	P14	ADSVCC	I	ADSVCC+0.3	ΔΣ analog SchB		
AN013/DSAN3N/ P14_15	P14	ADSVCC	I	ADSVCC+0.3	ΔΣ analog SchB		
AN012/DSAN3P/ P14_14	P14	ADSVCC	I	ADSVCC+0.3	ΔΣ analog SchB		
AN011/DSAN2N/ P14_13	P14	ADSVCC	I	ADSVCC+0.3	ΔΣ analog SchB		
AN010/DSAN2P/ P14_12	P14	ADSVCC	I	ADSVCC+0.3	ΔΣ analog SchB		
AN003/ DSAN1N	—	ADSVCC	I	ADSVCC+0.3	ΔΣ analog		
AN002/ DSAN1P	—	ADSVCC	I	ADSVCC+0.3	ΔΣ analog		
AN001/ DSAN0N	—	ADSVCC	I	ADSVCC+0.3	ΔΣ analog		

Table 37.2 Relationship between Power Name and Pin (2/7)

Pin Name (Value after Reset)	Port Group	Circuit Power Name	I/O	Maximum Input Voltage (V)	Input Buffer Type	Output Driving Ability	Note
AN000/ DSAN0P	—	ADSVCC	I	ADSVCC+0.3	ΔΣ analog		
P10_0	P10	EVCC	IO	EVCC+0.3	SchB	√	
P10_1	P10	EVCC	IO	EVCC+0.3	SchB	√	
P10_2	P10	EVCC	IO	EVCC+0.3	SchB	√	
P10_3	P10	EVCC	IO	EVCC+0.3	SchB	√	
P10_4	P10	EVCC	IO	EVCC+0.3	SchB	√	
P8_0	P8	EVCC	IO	EVCC+0.3	SchB	√	
P8_1	P8	EVCC	IO	EVCC+0.3	SchB	√	
P8_2	P8	EVCC	IO	EVCC+0.3	SchB	√	
P8_3	P8	EVCC	IO	EVCC+0.3	SchB	√	
P8_4	P8	EVCC	IO	EVCC+0.3	SchB	√	
P8_5	P8	EVCC	IO	EVCC+0.3	SchB	√	
P8_6	P8	EVCC	IO	EVCC+0.3	SchB	√	
P8_7	P8	EVCC	IO	EVCC+0.3	SchB	√	
P6_0	P6	EVCC	IO	EVCC+0.3	SchB	√	
P6_1	P6	EVCC	IO	EVCC+0.3	SchB	√	
P6_2	P6	EVCC	IO	EVCC+0.3	SchB	√	
P6_3	P6	EVCC	IO	EVCC+0.3	SchB	√	
P6_4	P6	EVCC	IO	EVCC+0.3	SchB	√	
P6_5	P6	EVCC	IO	EVCC+0.3	SchB	√	
P6_6	P6	EVCC	IO	EVCC+0.3	SchB	√	
P6_7	P6	EVCC	IO	EVCC+0.3	SchB	√	
P6_8	P6	EVCC	IO	EVCC+0.3	SchB	√	
P6_9	P6	EVCC	IO	EVCC+0.3	SchB	√	
P6_10	P6	EVCC	IO	EVCC+0.3	SchB	√	
P6_11	P6	EVCC	IO	EVCC+0.3	SchB	√	
P5_0	P5	EVCC	IO	EVCC+0.3	SchB	√	
P5_1	P5	EVCC	IO	EVCC+0.3	SchB	√	
P5_2	P5	EVCC	IO	EVCC+0.3	SchB	√	
P5_3	P5	EVCC	IO	EVCC+0.3	SchB	√	
P5_4	P5	EVCC	IO	EVCC+0.3	SchB	√	
P5_5	P5	EVCC	IO	EVCC+0.3	SchB	√	
P5_6	P5	EVCC	IO	EVCC+0.3	SchB	√	
P5_7	P5	EVCC	IO	EVCC+0.3	SchB	√	
P5_8	P5	EVCC	IO	EVCC+0.3	SchB	√	
P5_9	P5	EVCC	IO	EVCC+0.3	SchB	√	
P4_0	P4	EVCC	IO	EVCC+0.3	SchB	√	
P4_1	P4	EVCC	IO	EVCC+0.3	SchB	√	
P4_2	P4	EVCC	IO	EVCC+0.3	SchB	√	
P4_3	P4	EVCC	IO	EVCC+0.3	SchB	√	
P4_4	P4	EVCC	IO	EVCC+0.3	SchB	√	
P4_5	P4	EVCC	IO	EVCC+0.3	SchB	√	

Table 37.2 Relationship between Power Name and Pin (3/7)

Pin Name (Value after Reset)	Port Group	Circuit Power Name	I/O	Maximum Input Voltage (V)	Input Buffer Type	Output Driving Ability	Note
P4_6	P4	EVCC	IO	EVCC+0.3	SchB	√	
P4_7	P4	EVCC	IO	EVCC+0.3	SchB	√	
P4_8	P4	EVCC	IO	EVCC+0.3	SchB	√	
P4_9	P4	EVCC	IO	EVCC+0.3	SchB	√	
P4_10	P4	EVCC	IO	EVCC+0.3	SchB	√	
P4_11	P4	EVCC	IO	EVCC+0.3	SchB	√	
P4_12	P4	EVCC	IO	EVCC+0.3	SchB	√	
P4_13	P4	EVCC	IO	EVCC+0.3	SchB	√	
P4_14	P4	EVCC	IO	EVCC+0.3	SchB	√	
P4_15	P4	EVCC	IO	EVCC+0.3	SchB	√	
P11_0	P11	EVCC	IO	EVCC+0.3	SchB	√	
P11_1	P11	EVCC	IO	EVCC+0.3	SchB	√	
P11_2	P11	EVCC	IO	EVCC+0.3	SchB	√	
P11_3	P11	EVCC	IO	EVCC+0.3	SchB	√	
P11_4	P11	EVCC	IO	EVCC+0.3	SchB	√	
P11_5	P11	EVCC	IO	EVCC+0.3	SchB	√	
P13_9	P13	TTLVCC	IO	TTLVCC+0.3	SchC	√	
P13_8	P13	TTLVCC	IO	TTLVCC+0.3	SchC	√	
P13_7	P13	TTLVCC	IO	TTLVCC+0.3	SchC	√	
P13_6	P13	TTLVCC	IO	TTLVCC+0.3	SchC	√	
P13_5	P13	TTLVCC	IO	TTLVCC+0.3	SchC	√	
RHSB1FCLN	P17	LVDVCC	O	—	—		LVDS clock output
RHSB1FCLP	P17	LVDVCC	O	—	—		LVDS clock output
RHSB1SON	P17	LVDVCC	O	—	—		LVDS data output
RHSB1SOP	P17	LVDVCC	O	—	—		LVDS data output
RHSB0FCLN	P17	LVDVCC	O	—	—		LVDS clock output
RHSB0FCLP	P17	LVDVCC	O	—	—		LVDS clock output
RHSB0SON	P17	LVDVCC	O	—	—		LVDS data output
RHSB0SOP	P17	LVDVCC	O	—	—		LVDS data output
P13_0	P13	TTLVCC	IO	TTLVCC+0.3	SchC	√	
P13_1	P13	TTLVCC	IO	TTLVCC+0.3	SchC	√	
P13_2	P13	TTLVCC	IO	TTLVCC+0.3	SchC	√	

Table 37.2 Relationship between Power Name and Pin (4/7)

Pin Name (Value after Reset)	Port Group	Circuit Power Name	I/O	Maximum Input Voltage (V)	Input Buffer Type	Output Driving Ability	Note
P13_3	P13	TTLVCC	IO	TTLVCC+0.3	SchC	√	
P13_4	P13	TTLVCC	IO	TTLVCC+0.3	SchC	√	
FLMODE	—	SYSVCC	I	5.5+0.3	SchA (5 V tol.)		
MD0	—	SYSVCC	I	5.5+0.3	SchA (5 V tol.)		
MD1	—	SYSVCC	I	5.5+0.3	SchA (5 V tol.)		
$\overline{\text{RES}}$	—	SYSVCC	I	5.5+0.3	SchA (5 V tol.)		
$\overline{\text{TRST}}$/ $\overline{\text{LPDRST}}$		SYSVCC	I	5.5+0.3	SchA (5 V tol.)		
TDO/ LPDO/ FLSCI3TX		VCC	O	VCC+0.3	TTL2		
TMS/$\overline{\text{EVTO}}$		VCC	IO	VCC+0.3	TTL2		
TCK/ LPDCLKI/ FLSCI3SCK		VCC	I	VCC+0.3	TTL2		
TDI/ LPDI/ FLSCI3RX		VCC	I	VCC+0.3	TTL2		
$\overline{\text{DRDY}}$/ LPDCLKO		VCC	O	VCC+0.3	TTL2		
XTAL	—	VCC	O				
EXTAL	—	VCC	I	VCC+0.3	CMOS		
$\overline{\text{EVTO}}$	—	VCC	O				
$\overline{\text{EVTI}}$	—	VCC	I	VCC+0.3	SchB		
$\overline{\text{AUDRST}}$		VCC	I	VCC+0.3	SchA		
AUDCK		VCC	I	VCC+0.3	TTL2		
$\overline{\text{AUDSYNC}}$		VCC	I	VCC+0.3	TTL2		
AUDATA3		VCC	IO	VCC+0.3	TTL2		
AUDATA2		VCC	IO	VCC+0.3	TTL2		
AUDATA1		VCC	IO	VCC+0.3	TTL2		
AUDATA0		VCC	IO	VCC+0.3	TTL2		
NMI		VCC	I	5.5+0.3	SchA (5 V tol.)		
$\overline{\text{ERROROUT\_M}}$	—	VCC	O				
CK	—	EVCC	O				
P0_0	P0	EVCC	IO	EVCC+0.3	SchB	√	
P0_1	P0	EVCC	IO	EVCC+0.3	SchB	√	
P0_2	P0	EVCC	IO	EVCC+0.3	SchB	√	
P0_3	P0	EVCC	IO	EVCC+0.3	SchB	√	
P0_4	P0	EVCC	IO	EVCC+0.3	SchB	√	
P0_5	P0	EVCC	IO	EVCC+0.3	SchB	√	
P0_6	P0	EVCC	IO	EVCC+0.3	SchB	√	
P0_7	P0	EVCC	IO	EVCC+0.3	SchB	√	
P0_8	P0	EVCC	IO	EVCC+0.3	SchB	√	
P0_9	P0	EVCC	IO	EVCC+0.3	SchB	√	
P0_10	P0	EVCC	IO	EVCC+0.3	SchB	√	
P0_11	P0	EVCC	IO	EVCC+0.3	SchB	√	
P0_12	P0	EVCC	IO	EVCC+0.3	SchB	√	
P0_13	P0	EVCC	IO	EVCC+0.3	SchB	√	

Table 37.2 Relationship between Power Name and Pin (5/7)

Pin Name (Value after Reset)	Port Group	Circuit Power Name	I/O	Maximum Input Voltage (V)	Input Buffer Type	Output Driving Ability	Note
P0_14	P0	EVCC	IO	EVCC+0.3	SchB	√	
P1_0	P1	EVCC	IO	EVCC+0.3	SchB	√	
P1_1	P1	EVCC	IO	EVCC+0.3	SchB	√	
P1_2	P1	EVCC	IO	EVCC+0.3	SchB	√	
P1_3	P1	EVCC	IO	EVCC+0.3	SchB	√	
P1_4	P1	EVCC	IO	EVCC+0.3	SchB	√	
P1_5	P1	EVCC	IO	EVCC+0.3	SchB	√	
P1_6	P1	EVCC	IO	EVCC+0.3	SchB	√	
P1_7	P1	EVCC	IO	EVCC+0.3	SchB	√	
P1_8	P1	EVCC	IO	EVCC+0.3	SchB	√	
P1_9	P1	EVCC	IO	EVCC+0.3	SchB	√	
P1_10	P1	EVCC	IO	EVCC+0.3	SchB	√	
P1_11	P1	EVCC	IO	EVCC+0.3	SchB	√	
P1_12	P1	EVCC	IO	EVCC+0.3	SchB	√	
P1_13	P1	EVCC	IO	EVCC+0.3	SchB	√	
P1_14	P1	EVCC	IO	EVCC+0.3	SchB	√	
P1_15	P1	EVCC	IO	EVCC+0.3	SchB	√	
P2_0	P2	EVCC	IO	EVCC+0.3	SchB	√	
P2_1	P2	EVCC	IO	EVCC+0.3	SchB	√	
P2_2	P2	EVCC	IO	EVCC+0.3	SchB	√	
P2_3	P2	EVCC	IO	EVCC+0.3	SchB	√	
P2_4	P2	EVCC	IO	EVCC+0.3	SchB	√	
P2_5	P2	EVCC	IO	EVCC+0.3	SchB	√	
P2_6	P2	EVCC	IO	EVCC+0.3	SchB	√	
P2_7	P2	EVCC	IO	EVCC+0.3	SchB	√	
P3_0	P3	EVCC	IO	EVCC+0.3	SchB	√	
P3_1	P3	EVCC	IO	EVCC+0.3	SchB	√	
P3_2	P3	EVCC	IO	EVCC+0.3	SchB	√	
P3_3	P3	EVCC	IO	EVCC+0.3	SchB	√	
P3_4	P3	EVCC	IO	EVCC+0.3	SchB	√	
P3_5	P3	EVCC	IO	EVCC+0.3	SchB	√	
P3_6	P3	EVCC	IO	EVCC+0.3	SchB	√	
P3_7	P3	EVCC	IO	EVCC+0.3	SchB	√	
P7_0	P7	EVCC	IO	EVCC+0.3	SchB	√	
P7_1	P7	EVCC	IO	EVCC+0.3	SchB	√	
P7_2	P7	EVCC	IO	EVCC+0.3	SchB	√	
P7_3	P7	EVCC	IO	EVCC+0.3	SchB	√	
P7_4	P7	EVCC	IO	EVCC+0.3	SchB	√	
P7_5	P7	EVCC	IO	EVCC+0.3	SchB	√	
P7_6	P7	EVCC	IO	EVCC+0.3	SchB	√	
P7_7	P7	EVCC	IO	EVCC+0.3	SchB	√	
P7_8	P7	EVCC	IO	EVCC+0.3	SchB	√	
P7_9	P7	EVCC	IO	EVCC+0.3	SchB	√	

Table 37.2 Relationship between Power Name and Pin (6/7)

Pin Name (Value after Reset)	Port Group	Circuit Power Name	I/O	Maximum Input Voltage (V)	Input Buffer Type	Output Driving Ability	Note
P7_10	P7	EVCC	IO	EVCC+0.3	SchB	√	
P7_11	P7	EVCC	IO	EVCC+0.3	SchB	√	
AN172/P16_8	P16	A1VCC	I	A1VCC+0.3	SAR analog SchB		
AN171/P16_5	P16	A1VCC	I	A1VCC+0.3	SAR analog SchB		
AN170/P16_4	P16	A1VCC	I	A1VCC+0.3	SAR analog SchB		
AN162/P16_7	P16	A1VCC	I	A1VCC+0.3	SAR analog SchB		
AN161/P16_3	P16	A1VCC	I	A1VCC+0.3	SAR analog SchB		
AN160/P16_2	P16	A1VCC	I	A1VCC+0.3	SAR analog SchB		
AN152/P16_6	P16	A1VCC	I	A1VCC+0.3	SAR analog SchB		
AN151/P16_1	P16	A1VCC	I	A1VCC+0.3	SAR analog SchB		
AN150/P16_0	P16	A1VCC	I	A1VCC+0.3	SAR analog SchB		
AN142	—	A1VCC	I	A1VCC+0.3	SAR analog		
AN141	—	A1VCC	I	A1VCC+0.3	SAR analog		
AN140	—	A1VCC	I	A1VCC+0.3	SAR analog		
AN132	—	A1VCC	I	A1VCC+0.3	SAR analog		
AN131	—	A1VCC	I	A1VCC+0.3	SAR analog		
AN130	—	A1VCC	I	A1VCC+0.3	SAR analog		
AN122	—	A1VCC	I	A1VCC+0.3	SAR analog		
AN121	—	A1VCC	I	A1VCC+0.3	SAR analog		
AN120	—	A1VCC	I	A1VCC+0.3	SAR analog		
AN112	—	A1VCC	I	A1VCC+0.3	SAR analog		
AN111	—	A1VCC	I	A1VCC+0.3	SAR analog		
A0VCC							
A1VCC							
ADSVCC							
ADSVCL							ΔΣADC external capacitor pin
A0VREFH							
A1VREFH							
ADSVREFH							
ADSVREFL							
A0VSS							
A1VSS							
A1VSS (N.C.)							
ADSVSS							
PLLVCC							
PLLVSS							
EVCC							
EPTVOUT		VCC	O				
VCC							
LVDVCC							
LVDVSS							

Table 37.2 Relationship between Power Name and Pin (7/7)

Pin Name (Value after Reset)	Port Group	Circuit Power Name	I/O	Maximum Input Voltage (V)	Input Buffer Type	Output Driving Ability	Note
SYSVCC							
TTLVCC*1							
VDD							
VDD (N.C.)							
RAMVCL							
VSS							
VSS (N.C.)							

Note 1. The ground for TTLVCC is LVDVSS (the same as the ground for LVDVCC).

Note 2. Though the pins listed with the names of power supply pins (and as N.C.) do not affect operation of the microcontroller even if they are open-circuit, we recommend connecting the pins to the power supply of the same name as those that do not include N.C. to ensure that the power supply is stable. In addition, be sure to mount solder balls on the substrate. The pins listed with the names of power supply pins (and as N.C.) are internally connected with the power supply of the same name as those that do not include N.C.

37.2.2 Operating Conditions

Table 37.3 Operating Conditions

Symbol	Min.	Typ.	Max.	Unit
SYSVCC	3.0	3.3	3.6	V
VCC	3.0	3.3	3.6	V
PLLVCC	3.0	3.3	3.6	V
VDD	1.15	1.25	1.35	V
EVCC	4.5	5.0	5.5	V
LVDVCC	3.0	3.3	3.6	V
TTLVCC	3.0	3.3	3.6	V
	4.5	5.0	5.5	V
A0VCC*3, A1VCC	4.5	5.0	5.5	V
ADSVCC*3	4.5	5.0	5.5	V
A0VREFH, A1VREFH *1	4.5	5.0	5.5	V
ADSVREFH*2	4.5	5.0	5.5	V

Note 1. Set a value less than A0VCC and A1VCC.

Note 2. Set a value less than ADSVCC.

Note 3. Operated under the condition of A0VCC = ADSVCC.

Note: Supply the specified voltages to all power-supply voltage connections when starting operation. Turn off all power voltages when stopping operation. In power-off standby mode, turn off power voltages other than SYSVCC.

37.2.3 Input Voltage Characteristics

Table 37.4 DC Characteristics (Input Voltage)

Conditions: EVCC = 4.5 V to 5.5 V, TTLVCC = 4.5 V to 5.5 V or 3.0 V to 3.6 V, A0VCC, A1VCC = 4.5 V to 5.5 V,
A0VREFH = 4.5 V to A0VCC, A1VREFH = 4.5 V to A1VCC, ADSVCC = 4.5 V to 5.5 V,
ADSVREFH = 4.5 V to ADSVCC, PLLVCC = 3.0 V to 3.6 V, SYSVCC = 3.0 V to 3.6 V,
VCC = 3.0 V to 3.6 V, LVDVCC = 3.0 V to 3.6 V, VDD = 1.15 V to 1.35 V,
VSS = PLLVSS = A0VSS = A1VSS = ADSVSS = ADSVREFL = LVDVSS = 0 V
Tj = -40°C to 150°C

Item	Symbol	Min.	Typ.	Max.	Unit	Measurement Condition
Schmitt trigger input voltage (Buffer type A)	V_T^+ (V_{IH})	SYSVCC × 0.75 VCC × 0.75	—	5.5 + 0.3*1 VCC + 0.3*2	V	See **Table 37.2** (Item of SchA input buffer type)
	V_T^- (V_{IL})	-0.3	—	SYSVCC × 0.25 VCC × 0.25	V	
	V_{HS}	SYSVCC × 0.2 VCC × 0.2			V	
Schmitt trigger input voltage (Buffer type B)	V_T^+ (V_{IH})	EVCC × 0.7 A0VCC × 0.7 A1VCC × 0.7 ADSVCC × 0.7 VCC × 0.74	—	EVCC + 0.3 A0VCC + 0.3 A1VCC + 0.3 ADSVCC + 0.3 VCC + 0.3	V	See **Table 37.2** (Item of SchB input buffer type)
	V_T^- (V_{IL})	-0.3	—	EVCC × 0.42 A0VCC × 0.42 A1VCC × 0.42 ADSVCC × 0.42 VCC × 0.34	V	
	V_{HS}	EVCC × 0.082 A0VCC × 0.082 A1VCC × 0.082 ADSVCC × 0.082 VCC × 0.082	—	—	V	
Schmitt trigger input voltage (Buffer type C)	V_T^+ (V_{IH})	TTLVCC × 0.6	—	TTLVCC + 0.3	V	See **Table 37.2** (Item of SchC input buffer type) TTLVCC = 3.0 to 3.6 V, 4.5 to 5.5 V
	V_T^- (V_{IL})	-0.3	—	TTLVCC × 0.36	V	
	V_{HS}	TTLVCC × 0.082	—	—	V	
TTL input voltage	V_{IH}	2.2	—	VCC + 0.3	V	See **Table 37.2** (Item of TTL2 input buffer type)
	V_{IL}	-0.3	—	0.8	V	
Clock input pin voltage (EXTAL)	V_{IH}	VCC × 0.7	—	VCC + 0.3	V	
	V_{IL}	-0.3	—	VCC × 0.2	V	

Note 1. $\overline{\text{AUDRST}}$ is excluded.
Note 2. $\overline{\text{AUDRST}}$

37.2.4 Input Leak Current Characteristics

Table 37.5 DC Characteristics (Input Leak Current)

Conditions: EVCC = 4.5 V to 5.5 V, TTLVCC = 4.5 V to 5.5 V or 3.0 V to 3.6 V, A0VCC, A1VCC = 4.5 V to 5.5 V,
A0VREFH = 4.5 V to A0VCC, A1VREFH = 4.5 V to A1VCC, ADSVCC = 4.5 V to 5.5 V,
ADSVREFH = 4.5 V to ADSVCC, PLLVCC = 3.0 V to 3.6 V, SYSVCC = 3.0 V to 3.6 V,
VCC = 3.0 V to 3.6 V, LVDVCC = 3.0 V to 3.6 V, VDD = 1.15 V to 1.35 V,
VSS = PLLVSS = A0VSS = A1VSS = ADSVSS = ADSVREFL = LVDVSS = 0 V
Tj = -40°C to 150°C

Item		Symbol	Min.	Typ.	Max.	Unit	Measurement Condition
Input leak current	Other than A/D port	| Iin |	—	—	1	µA	Vin = 0 V to EVCC Vin = 0 V to TTLVCC Vin = 0 V to SYSVCC, Vin = 0 V to VCC
	A/D port	| Iin |	—	—	0.1	µA	Vin = 0 V to A0VCC, A1VCC Vin = 0 V to ADSVCC

37.2.5 Pull-Up/Pull-Down MOS Current Characteristics

Table 37.6 DC Characteristics (Pull-Up/Pull-Down MOS Current)

Conditions: EVCC = 4.5 V to 5.5 V, TTLVCC = 4.5 V to 5.5 V or 3.0 V to 3.6 V, A0VCC, A1VCC = 4.5 V to 5.5 V,
A0VREFH = 4.5 V to A0VCC, A1VREFH = 4.5 V to A1VCC, ADSVCC = 4.5 V to 5.5 V,
ADSVREFH = 4.5 V to ADSVCC, PLLVCC = 3.0 V to 3.6 V, SYSVCC = 3.0 V to 3.6 V,
VCC = 3.0 V to 3.6 V, LVDVCC = 3.0 V to 3.6 V, VDD = 1.15 V to 1.35 V,
VSS = PLLVSS = A0VSS = A1VSS = ADSVSS = ADSVREFL = LVDVSS = 0 V
Tj = -40°C to 150°C

Item		Symbol	Min.	Typ.	Max.	Unit	Measurement Condition
Input pull-up MOS current	TMS, TDI, TCK (Including the case of use as JTAG port)	$-I_{pu}$	—	—	350	µA	Vin = 0 V, VCC = 3.0 to 3.6 V
	$\overline{\text{AUDCK}}$, $\overline{\text{AUDSYNC}}$ (VCC power pin)		—	—	350	µA	Vin = 0 V VCC = 3.0 to 3.6 V
	EVCC power pin		—	—	350	µA	Vin = 0 V EVCC = 4.5 to 5.5 V
	TTLVCC power pin		15	—	90	µA	Vin = 0 V, TTLVCC = 3.0 to 3.6 V
			40	—	190	µA	Vin = 0 V, TTLVCC = 4.5 to 5.5 V
Pull-up MOS current	AUDATA3-0[*1]		—	—	350	µA	Vin = 0 V VCC = 3.0 to 3.6 V
Input pull-down MOS current	$\overline{\text{RES}}$	I_{pd}	25	60	120	µA	Vin = SYSVCC, SYSVCC = 3.6 V
			5	20	40	µA	Vin = SYSVCC, SYSVCC = 2.0 V
	$\overline{\text{TRST}}$		—	—	350	µA	Vin = SYSVCC, SYSVCC = 3.0 to 3.6 V
	FLMODE, MD0, MD1 (SYSVCC power pin)		15	—	120	µA	Vin = SYSVCC, SYSVCC = 3.0 to 3.6 V
	$\overline{\text{AUDRST}}$		—	—	350	µA	Vin = VCC, VCC = 3.0 to 3.6 V
	NMI		15	—	120	µA	Vin = VCC, VCC = 3.0 to 3.6 V
	EVCC power pin		—	—	350	µA	Vin = EVCC, EVCC = 4.5 to 5.5 V
	TTLVCC power pin		15	—	120	µA	Vin = TTLVCC, TTLVCC = 3.0 to 3.6 V
			50	—	240	µA	Vin = TTLVCC TTLVCC = 4.5 V to 5.5 V

Note 1. The pull-up of AUDATA3-0 is valid not only in input but also in output.

37.2.6 Output Voltage Characteristics

Table 37.7 DC Characteristics (Output Voltage)

Conditions: EVCC = 4.5 V to 5.5 V, TTLVCC = 4.5 V to 5.5 V or 3.0 V to 3.6 V, A0VCC, A1VCC = 4.5 V to 5.5 V,
A0VREFH = 4.5 V to A0VCC, A1VREFH = 4.5 V to A1VCC, ADSVCC = 4.5 V to 5.5 V,
ADSVREFH = 4.5 V to ADSVCC, PLLVCC = 3.0 V to 3.6 V, SYSVCC = 3.0 V to 3.6 V,
VCC = 3.0 V to 3.6 V, LVDVCC = 3.0 V to 3.6 V, VDD = 1.15 V to 1.35 V,
VSS = PLLVSS = A0VSS = A1VSS = ADSVSS = ADSVREFL = LVDVSS = 0 V
Tj = -40°C to 150°C

Item		Symbol	Min.	Typ.	Max.	Unit	Measurement Condition
Output high level voltage	EVCC power pin	V_{OH}	EVCC - 0.5	—	—	V	I_{OH} = 200 µA, EVCC = 4.5 V to 5.5 V
			EVCC - 1.0	—	—	V	I_{OH} = 1mA, EVCC = 4.5 V to 5.5 V
	TTLVCC power pin		TTLVCC - 0.4	—	—	V	I_{OH} = 2mA, TTLVCC= 4.5 V to 5.5 V
			TTLVCC - 0.4	—	—	V	I_{OH} = 2mA, TTLVCC= 3.0 V to 3.6 V
	VCC power pin		VCC - 0.1	—	—	V	I_{OH} = 50 µA, VCC = 3.0 V to 3.6 V
			VCC - 1.0	—	—	V	I_{OH} = 200 µA, VCC = 3.0 V to 3.6 V
Output low level voltage	EVCC power pin	V_{OL}	—	—	0.4	V	I_{OL} = 1.6 mA EVCC = 4.5 V to 5.5 V
			—	—	1.2	V	I_{OL} = 4 mA EVCC = 4.5 V to 5.5 V
	TTLVCC power pin		—	—	0.4	V	I_{OL} = 2mA, TTLVCC = 4.5 V to 5.5 V
			—	—	0.4	V	I_{OL} = 2mA, TTLVCC = 3.0 V to 3.6 V
	VCC power pin		—	—	0.1	V	I_{OL} = 50 µA VCC = 3.0 V to 3.6 V
			—	—	0.4	V	I_{OL} = 1.6 mA VCC = 3.0 V to 3.6 V

37.2.7 Allowable Output Current

Table 37.8 DC Characteristics (Allowable Output Current)

Conditions: EVCC = 4.5 V to 5.5 V, TTLVCC = 4.5 V to 5.5 V or 3.0 V to 3.6 V, A0VCC, A1VCC = 4.5 V to 5.5 V,
A0VREFH = 4.5 V to A0VCC, A1VREFH = 4.5 V to A1VCC, ADSVCC = 4.5 V to 5.5 V,
ADSVREFH = 4.5 V to ADSVCC, PLLVCC = 3.0 V to 3.6 V, SYSVCC = 3.0 V to 3.6 V,
VCC = 3.0 V to 3.6 V, LVDVCC = 3.0 V to 3.6 V, VDD = 1.15 V to 1.35 V,
VSS = PLLVSS = A0VSS = A1VSS = ADSVSS = ADSVREFL = LVDVSS = 0 V
Tj = -40°C to 150°C

Item	Symbol	Min.	Typ.	Max.	Unit
Output low-level allowable current (per pin)	I_{OL}	—	—	4.0	mA
Output low-level allowable current (total)	ΣI_{OL}	—	—	80	mA
Output high-level allowable current (per pin)	I_{OH}	—	—	2.0	mA
Output high-level allowable current (total)	ΣI_{OH}	—	—	25	mA

37.2.8 Injection Current

Table 37.9 DC Characteristics (Injection Current)

Conditions: EVCC = 4.5 V to 5.5 V, TTLVCC = 4.5 V to 5.5 V or 3.0 V to 3.6 V, A0VCC, A1VCC = 4.5 V to 5.5 V,
A0VREFH = 4.5 V to A0VCC, A1VREFH = 4.5 V to A1VCC, ADSVCC = 4.5 V to 5.5 V,
ADSVREFH = 4.5 V to ADSVCC, PLLVCC = 3.0 V to 3.6 V, SYSVCC = 3.0 V to 3.6 V,
VCC = 3.0 V to 3.6 V, LVDVCC = 3.0 V to 3.6 V, VDD = 1.15 V to 1.35 V,
VSS = PLLVSS = A0VSS = A1VSS = ADSVSS = ADSVREFL = LVDVSS = 0 V
Tj = -40°C to 150°C

Item		Symbol	Min.	Typ.	Max.	Unit
DC injection current (per pin)	Logic pin	I_{IC}	-2.0	—	2.0	mA
	Analog pin		-3.0	—	3.0	mA
DC injection current (total)		$\Sigma \mid I_{IC} \mid$	-50.0	—	50.0	mA

37.2.9 LVDS Driver

Table 37.10 DC Characteristics (LVDS Driver Characteristics)

Conditions: EVCC = 4.5 V to 5.5 V, TTLVCC = 4.5 V to 5.5 V or 3.0 V to 3.6 V, A0VCC, A1VCC = 4.5 V to 5.5 V,
A0VREFH = 4.5 V to A0VCC, A1VREFH = 4.5 V to A1VCC, ADSVCC = 4.5 V to 5.5 V,
ADSVREFH = 4.5 V to ADSVCC, PLLVCC = 3.0 V to 3.6 V, SYSVCC = 3.0 V to 3.6 V,
VCC = 3.0 V to 3.6 V, LVDVCC = 3.0 V to 3.6 V, VDD = 1.15 V to 1.35 V,
VSS = PLLVSS = A0VSS = A1VSS = ADSVSS = ADSVREFL = LVDVSS = 0 V
Tj = -40°C to 150°C

Item	Symbol	Min.	Typ.	Max.	Unit	Measurement Condition
Output high-level voltage	V_{OH}	—	—	1525	mV	**Figure 37.1**, **Figure 37.2**
Output low-level voltage	V_{OL}	875	—	—	mV	
Output differential voltage	V_{OD}	150	—	400	mV	
Offset voltage	V_{OS}	1075	—	1325	mV	
Output impedance	R0	40	—	140	Ω	

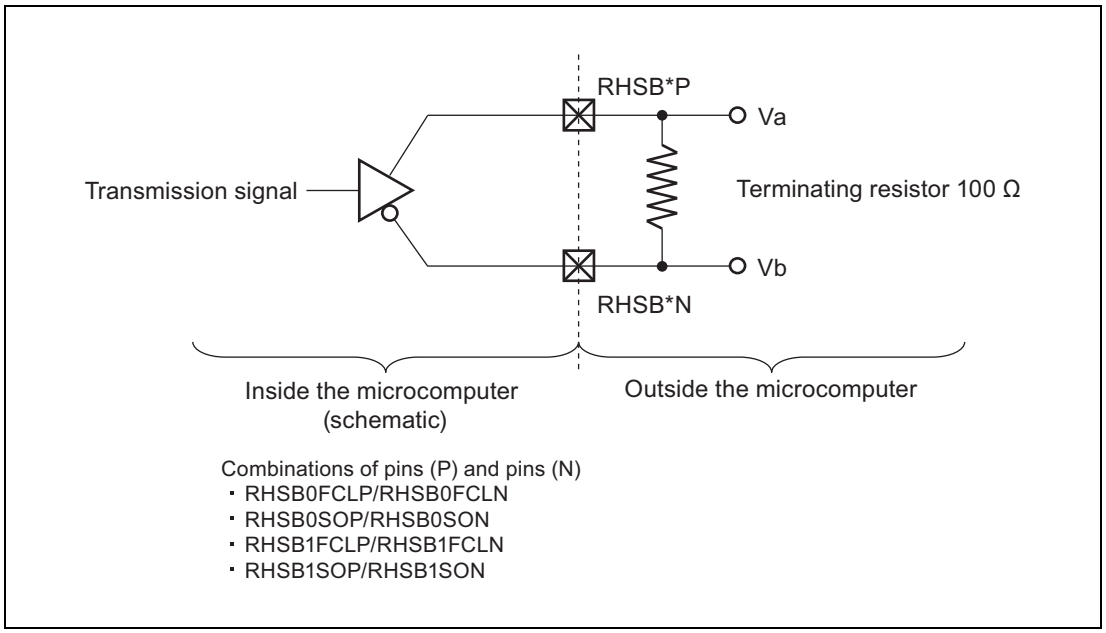


Figure 37.1 LVDS Driver Va/Vb Measurement Conditions

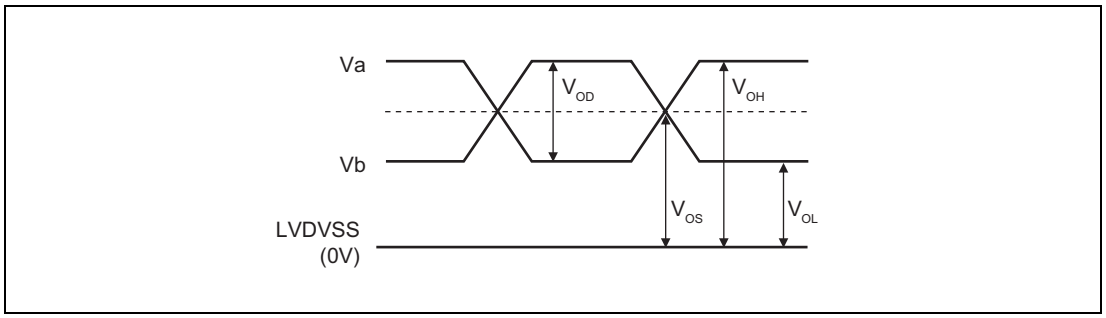


Figure 37.2 Meaning of LVDS Driver Symbols

37.2.10 Input Capacitance

Table 37.11 DC Characteristics (Input Capacitance)

Conditions: EVCC = 4.5 V to 5.5 V, TTLVCC = 4.5 V to 5.5 V or 3.0 V to 3.6 V, A0VCC, A1VCC = 4.5 V to 5.5 V, A0VREFH = 4.5 V to A0VCC, A1VREFH = 4.5 V to A1VCC, ADSVCC = 4.5 V to 5.5 V, ADSVREFH = 4.5 V to ADSVCC, PLLVCC = 3.0 V to 3.6 V, SYSVCC = 3.0 V to 3.6 V, VCC = 3.0 V to 3.6 V, LVDVCC = 3.0 V to 3.6 V, VDD = 1.15 V to 1.35 V, VSS = PLLVSS = A0VSS = A1VSS = ADSVSS = ADSVREFL = LVDVSS = 0 V

Item		Symbol	Min.	Typ.	Max.	Unit	Measurement Condition
Input capacitance	All pins	C_{in}	—	10	20	pF	Vin = 0 V, f = 1 MHz, Tj = 25°C

37.2.11 Supply Current Characteristics

Table 37.12 DC Characteristics (Supply Current)

Conditions: EVCC = 4.5 V to 5.5 V, TTLVCC = 4.5 V to 5.5 V or 3.0 V to 3.6 V, A0VCC, A1VCC = 4.5 V to 5.5 V,
A0VREFH = 4.5 V to A0VCC, A1VREFH = 4.5 V to A1VCC, ADSVCC = 4.5 V to 5.5 V,
ADSVREFH = 4.5 V to ADSVCC, PLLVCC = 3.0 V to 3.6 V, SYSVCC = 3.0 V to 3.6 V,
VCC = 3.0 V to 3.6 V, LVDVCC = 3.0 V to 3.6 V, VDD = 1.15 V to 1.35 V,
VSS = PLLVSS = A0VSS = A1VSS = ADSVSS = ADSVREFL = LVDVSS = 0 V
Tj = -40°C to 150°C

Item		Symbol	Min.	Typ.*1	Max.	Unit	Measurement Condition
Core supply current (VDD power)	Normal operation	Idd	—	230	420	mA	Operation at 320 MHz
			—	180	390	mA	Operation at 240 MHz
	Current during reset	Iddrst	10	70	200	mA	
VCC supply current (when EPT is not used)*2	Normal operation (excluding erasure of code flash)	I_{CC}	—	10	23	mA	
	Erasure of code flash	Icc_cferase	—	20	40	mA	
	Current during reset	Iccrst	3	7	15	mA	
System supply current (SYSVCC)	Normal operation	I_{SYS}	—	0.25	1	mA	
	Current during reset	I_{sysrst}	0.1	0.3	1	mA	
PLL supply current (PLLVCC)		I_{PLL}	2	3.5	5	mA	
RHSB supply current (LVDVCC)		I_{LVDS}	—	14	24	mA	
Analog power current (A0VCC, A1VCC)		I_{AVCC}	—	3.5	6	mA	
Analog power current (ADSVCC)		I_{ADSVCC}	—	12	15	mA	
ADC reference power current (A0VREFH, A1VREFH)		I_{AVREF}	—	220	500	µA	
ADC reference power current (ADSVREFH)		I_{AVREF_DS}	—	400	1000	µA	

Note 1. At Tj = 25°C

Note 2. When an EPT is to be used, calculate the current through the EPT from its current gain and the current from EPTVOUT, then add the result to the value for VCC.

CAUTIONS

1. **When the A/D converter is not used or it is in the standby state, do not open the A0VCC pin, A1VCC pin, A0VREFH pin, A1VREFH pin, ADSVREFH pin, ADSVREFL pin, A0VSS pin, and A1VSS pin.**
2. **Supply current values are with all output pins unloaded when V_{IH}min = VCC – 0.5 V/EVCC – 0.5 V and V_{IL} = 0.5 V.**

37.2.12 Standby Current

Table 37.13 DC Characteristics (Standby)

Conditions: EVCC = 0 V to EVCCstdby1, SYSVCC = SYSVCCLow to 3.6 V,
Other power supplies 0 V to operating voltage,
VSS = PLLVSS = A0VSS = A1VSS = ADSVSS = ADSVREFL = LVDVSS = 0 V
Tj = -40°C to 150°C

Item		Symbol	Min.	Typ.	Max.	Unit	Measurement Condition
Supply current (SYSVCC)	Power-off standby	I_{SB}	—	—	0.75	mA	-40°C <Tj ≤ 50°C
			—	—	1	mA	50°C < Tj ≤ 105°C
			—	—	1.5	mA	105°C < Tj ≤ 150°C

37.2.13 Power Voltage Monitoring Characteristics

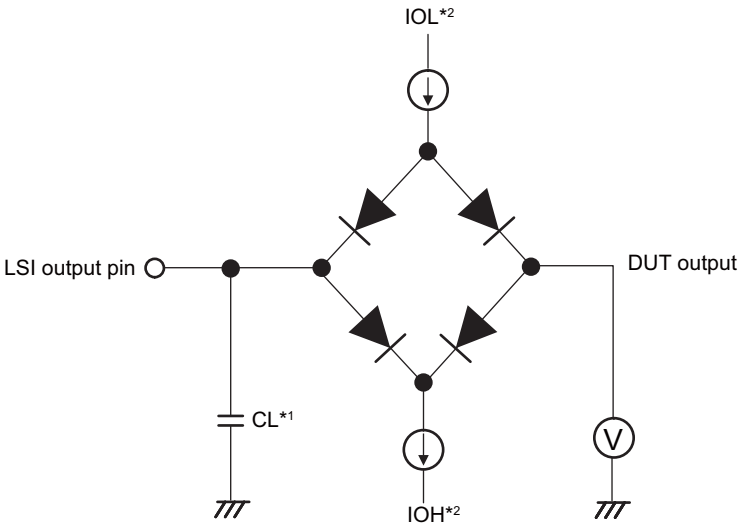
Table 37.14 DC Characteristics (Power Voltage Monitoring)

Conditions: EVCC = 4.5 V to 5.5 V, TTLVCC = 4.5 V to 5.5 V or 3.0 V to 3.6 V, A0VCC, A1VCC = 4.5 V to 5.5 V,
A0VREFH = 4.5 V to A0VCC, A1VREFH = 4.5 V to A1VCC, ADSVCC = 4.5 V to 5.5 V,
ADSVREFH = 4.5 V to ADSVCC, PLLVCC = 3.0 V to 3.6 V, SYSVCC = 3.0 V to 3.6 V,
VCC = 3.0 V to 3.6 V, LVDVCC = 3.0 V to 3.6 V,
VSS = PLLVSS = A0VSS = A1VSS = ADSVSS = ADSVREFL = LVDVSS = 0 V
Tj = -40°C to 150°C

Item			Symbol	Min.	Typ.	Max.	Unit	Measurement Condition
Power voltage monitoring upper-limit voltage	HDETCTL.VDDREFH 1/0 setting	00	—	1.40	1.50	1.60	V	
Power voltage monitoring lower-limit voltage	LDETCTL.VDDREFL 1/0 setting	00	—	0.85	0.95	1.05	V	

37.3 AC Characteristics

- Unless otherwise described, the following timing conditions are applied.
  SYSVCC = VCC = PLLVCC = LVDVCC = TTLVCC = 3.0 V to 3.6 V
  EVCC = 4.5 V to 5.5 V, A0VCC = A1VCC = 4.5 V to 5.5 V, A0VREFH = 4.5 V to A0VCC,
  A1VREFH = 4.5 V to A1VCC,
  ADSVCC = 4.5 V to 5.5 V, ADSVREFH = 4.5 V to ADSVCC, VDD = 1.15 to 1.35 V,
  VSS = PLLVSS = A0VSS = A1VSS = ADSVSS = ADSVREFL = LVDVSS = 0 V,
  Tj = -40°C to 150°C
- In the port control register, conditions where all output pins of the module used in the same channel are set to the same driving ability are applied to output pins whose driving ability is selectable. Unless otherwise specified, all driving ability settings are included.
- Unless otherwise described, AC measurement conditions described in **Figure 37.3** are applied.
  Input reference level High level: V_{IH} minimum value; Low level: V_{IL} maximum value
  Output reference level High level: 2.0 V; Low level: 0.8 V
  Input rising time, falling time: 1 ns



Notes: 1. CL is a total value including the capacitance of the measuring equipment.
Each pin is set as follows:
15 pF: FlexRay pin
30 pF: CK, NEXUS, JTAG, LDU, AUD RAM monitor pins
50 pF: Pins other than above
2. IOH = 200 uA, IOL = 1.6 mA

Figure 37.3 AC Measurement Conditions

37.3.1 Power On/Off Timings

(1) When EPT is not used

Table 37.15 Power On/Off Timings

Item	Symbol	Min.	Max.	Unit	Note	Reference
External reset L time at power-on	tRESW1	10	—	ms	*1	Figure 37.4
External reset L time at power-off	tRESW2	2	—	µs	*2	
PLL1 lock in time	tPLL1L0	—	1	ms	*3	

Note 1. tRESW1 is the reset time required for the supply of internal clock signals to become stable after all power supplies are turned on. No restriction applies to potential differences between each power supply during power up.

Note 2. tRESW2 is the time from assertion of the reset signal until all of the power voltages have dropped below the lower-limit voltages.

Note 3. tPLL1L0 is the time required for PLL1 to lock in after MOSC oscillation has become stable.

CAUTION

- **The states of I/O pins are not reset during the noise cancellation interval (max. 1.2 µs) of the reset signal following its assertion while power is being turned off. During that time, do not allow any input of mid-range potential to the pin or contention of output data.**
- **If power is disconnected during programming or erasure of flash memory, the flash memory may be destroyed.**

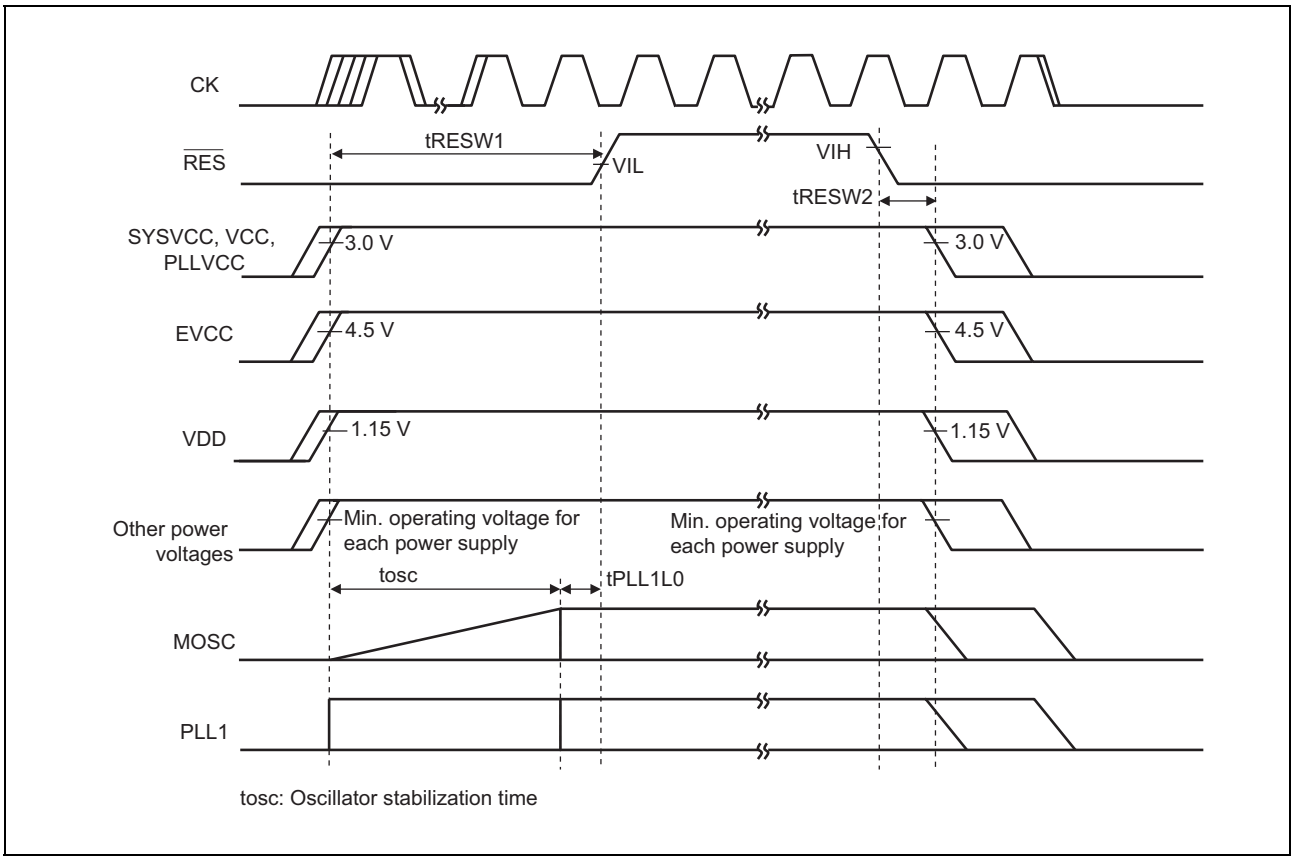


Figure 37.4 Power On/Off Timings When EPT is not Used

(2) When EPT is used

Table 37.16 Power On/Off Timings

Item	Symbol	Min.	Max.	Unit	Condition	Reference
External reset L time at power-on	tRESW4	10	—	ms	*1	**Figure 37.5**
External reset L time at power-off	tRESW5	2	—	µs	*2	
PPL1 lock-in time	tPLL1L0	—	1	ms	*3	

Note 1. tRESW4 is the reset time required for the supply of internal clock signals to become stable after all power supplies are turned on. No restriction applies to potential differences between each power supply during power up.

Note 2. tRESW5 is the time from assertion of the reset signal until all of the power voltages have dropped below the lower-limit voltages.

Note 3. tPLL1L0 is the time required for PLL1 to lock in after MOSC oscillation has become stable.

CAUTION

- **The states of I/O pins are not reset during the noise cancellation interval (max. 1.2 µs) of the reset signal following its assertion while power is being turned off. During that time, do not allow any input of mid-range potential to the pin or contention of output data.**
- **If power is disconnected during programming or erasure of flash memory, the flash memory may be destroyed.**

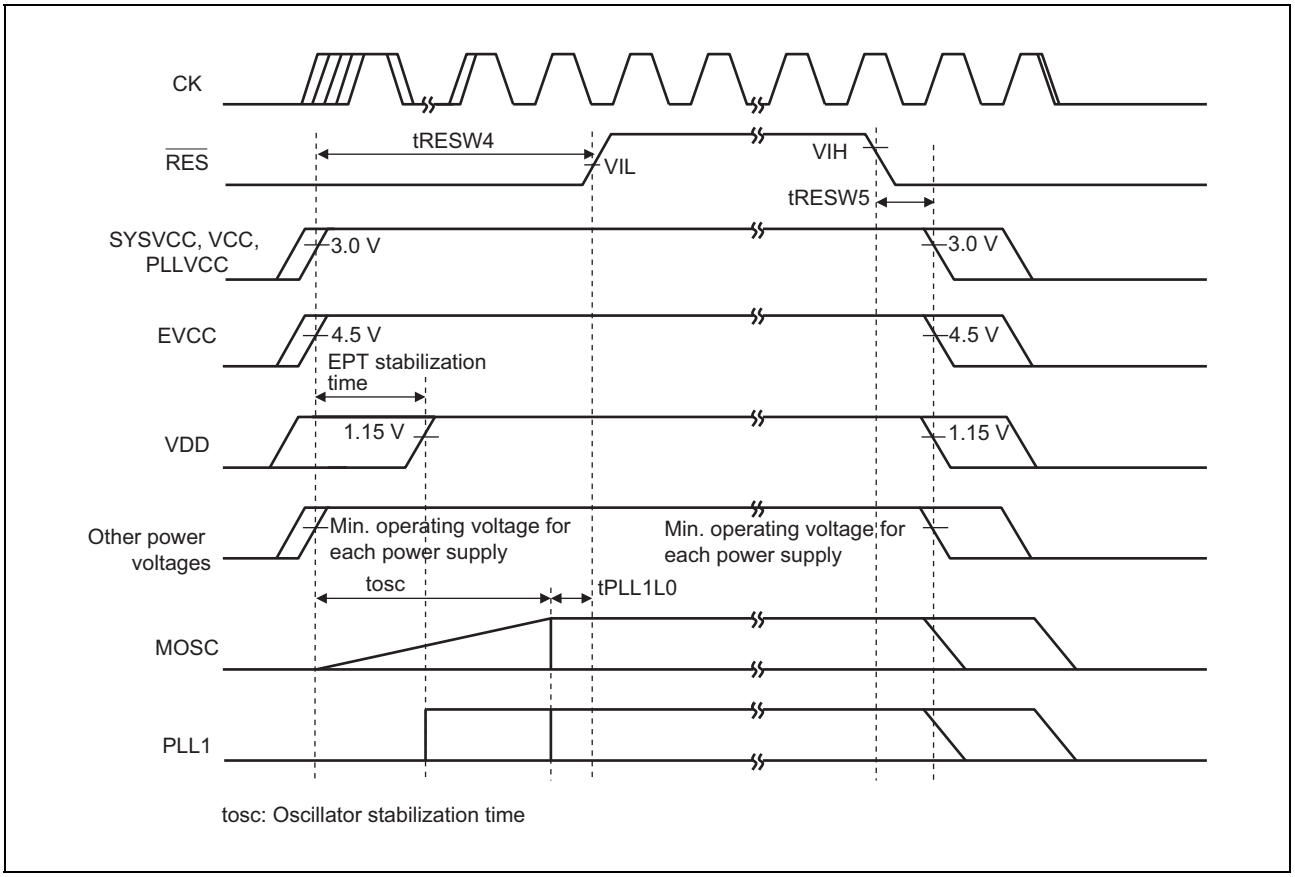


Figure 37.5 Power On/Off Timings when EPT is Used

37.3.2 Standby Transition/Return Timings

Table 37.17 Power-Off Standby Timing

Item	Symbol	Condition	Min.	Max.	Unit	Reference
Reset assertion time before VCC off	tVCC		2	—	µs	**Figure 37.6**
Reset assertion time before VDD off	tVDD		2	—	µs	
Reset assertion time before EVCC off	tEVCC		2	—	µs	
Reset assertion time before power off other than VCC, VDD, and EVCC	tVoltage		2	—	µs	
Reset hold time at return from power-off standby mode	tRESW3	*1	tRESW1	—	ms	
		*2	tRESW4		ms	
EVCC voltage at transition to power-off standby mode	EVCCstdby1	*3	2.1	—	V	
EVCC voltage at return from power-off standby mode	EVCCstdby2	*4	—	2.5	V	
SYSVCC voltage in power-off standby mode	SYSVCCLow		2.7	3.6	V	

Note 1. When EPT is not used.

Note 2. When EPT is used.

Note 3. This is the EVCC voltage for transition to power-off standby mode. This value must be equal to or less than the min. EVCCstdby1 value.

Note 4. This is the EVCC voltage for return from power-off standby mode. This value must be equal to or more than the max. EVCCstdby2 value.

CAUTION

For resetting while flash memory is being programmed or erased, follow the specifications in the *RH850/E1x Flash Memory User's Manual: Hardware Interface*.

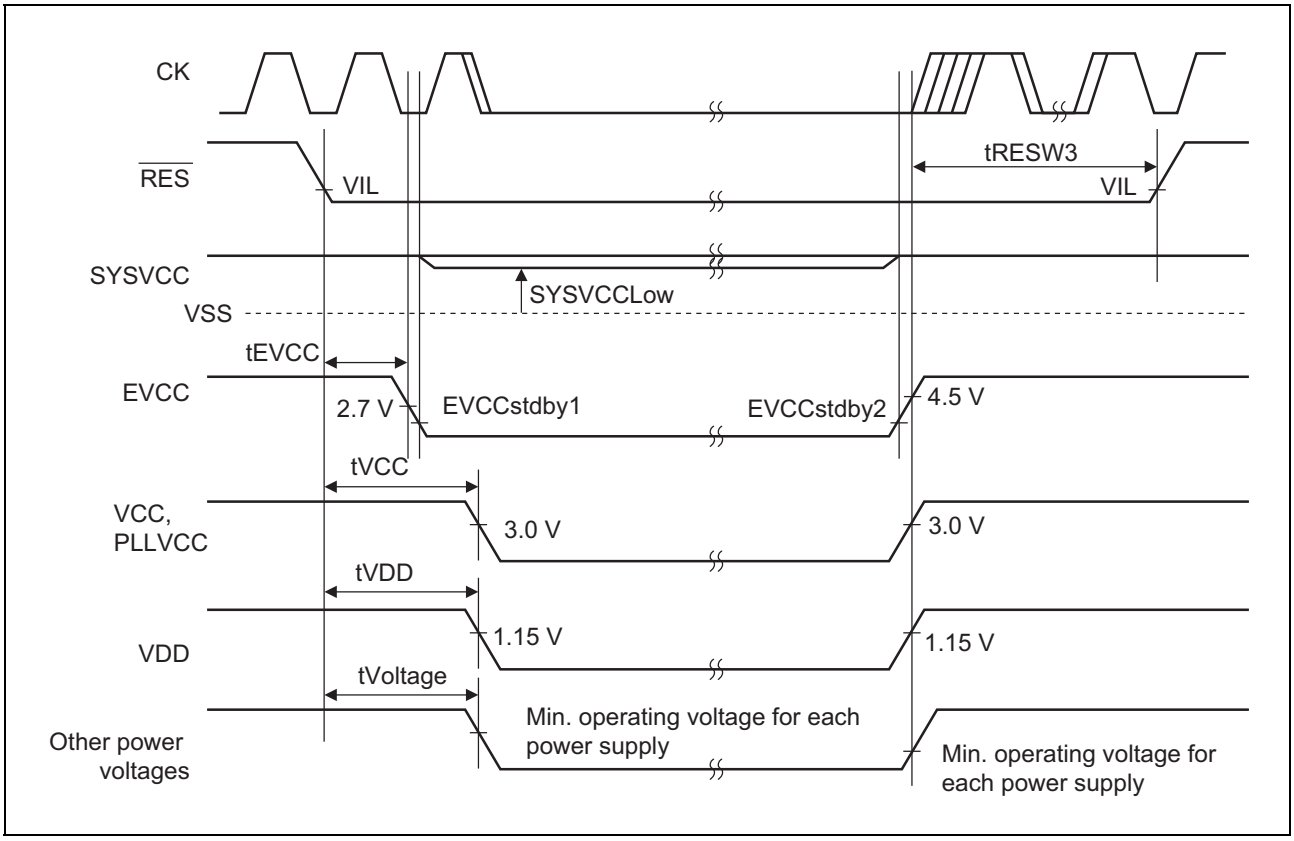


Figure 37.6 Power-Off Standby Timing

37.3.3 Clock Timing

37.3.3.1 External Clock and Output Clock

Table 37.18 Clock Timing

Item	Symbol	Min.	Typ.	Max.	Unit	Reference
EXTAL clock input frequency	f_{EX}	19.9	—	20	MHz	**Figure 37.7**
EXTAL clock input low-level pulse width	t_{EXL}	20	—	—	ns	
EXTAL clock input high-level pulse width	t_{EXH}	20	—	—	ns	
EXTAL clock input rising time	t_{EXR}	—	—	4	ns	
EXTAL clock input falling time	t_{EXF}	—	—	4	ns	
CK clock frequency*1	f_{op}	0.995	1	2	MHz	**Figure 37.8**
CK clock low-level pulse width	t_{CL}	230	—	—	ns	
CK clock high-level pulse width	t_{CH}	230	—	—	ns	
CK clock rising time*2	t_{CR}	—	—	15	ns	
CK clock falling time*2	t_{CF}	—	—	15	ns	

Note 1. The frequency of clock that is output from the CK pin is 1 MHz (when EXTAL input = 20 MHz), which is divided according to the setting by the baud rate generator (BRG). Clock jitter values depend on the board design.

Note 2. CL = 30pF

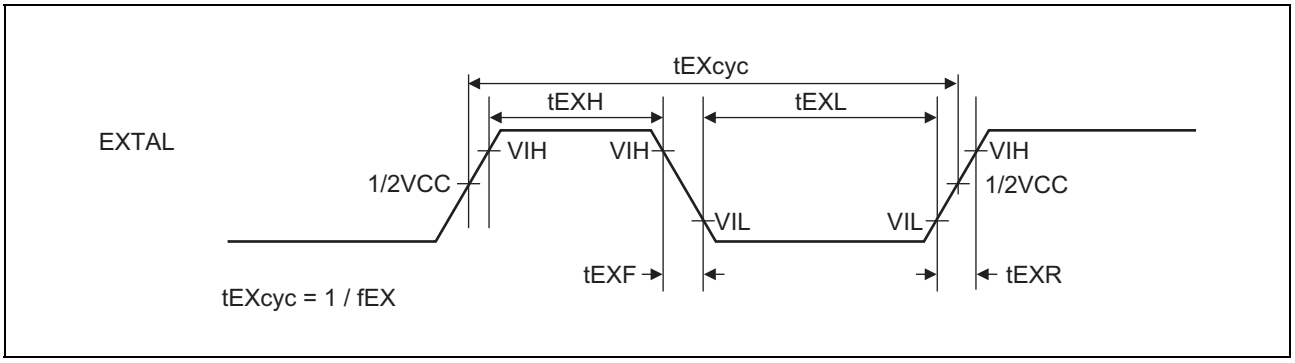


Figure 37.7 EXTAL Timing

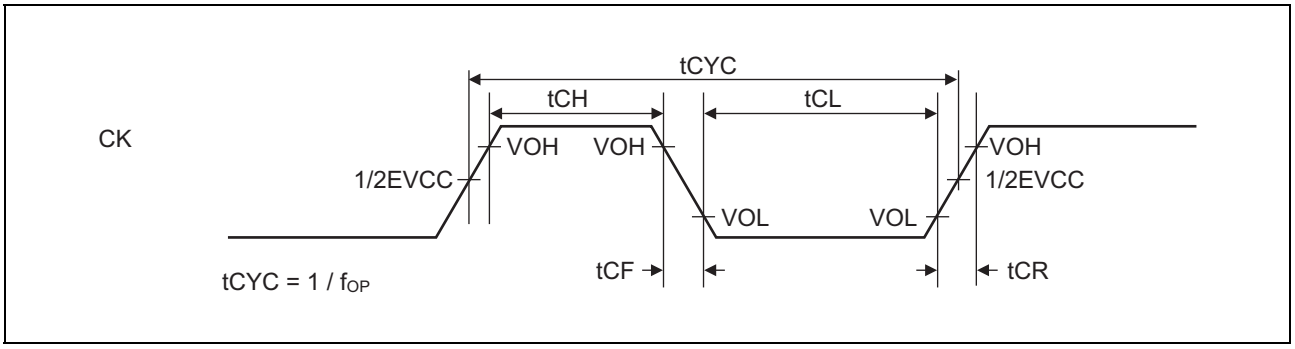


Figure 37.8 CK Timing

37.3.3.2 Spread Spectrum Clock Generator

Table 37.19 SSCG Timing

Item	Symbol	Min.	Typ.	Max.	Unit
Modulation frequency	fmod	20	—	100	kHz
Frequency dithering range[*1]	fdit	4.1	—	—	%
Frequency stabilization time (OFF → ON)		—	—	1.6	ms

Note 1. The modulation method is applied only to down spread.

37.3.3.3 Oscillation Frequency Accuracy of the On-Chip Oscillator

Table 37.20 Oscillation Frequency Accuracy of the On-Chip Oscillator

Item	Symbol	Min.	Typ.	Max.	Unit
ROSC oscillation frequency	fLOCO	160	240	360	kHz

37.3.4 Output Slew Rate

EVCC power supply pins

Table 37.21 Selection of Driving Ability = High

Item	Symbol	Condition	Min.	Typ.	Max.	Unit	Reference
Output rising time/falling time slew rate	tR, tF	CL = 25pF	—	4	6	ns	**Figure 37.9**
		CL = 50pF	—	6	12	ns	
		CL = 75pF	—	8	16	ns	
		CL = 100pF	—	10	20	ns	

Table 37.22 Selection of Driving Ability = Mid

Item	Symbol	Condition	Min.	Typ.	Max.	Unit	Reference
Output rising time/falling time slew rate	tR, tF	CL = 25pF	—	8	15	ns	**Figure 37.9**
		CL = 50pF	—	15	30	ns	
		CL = 75pF	—	23	45	ns	
		CL = 100pF	—	30	60	ns	

Table 37.23 Selection of Driving Ability = Low

Item	Symbol	Condition	Min.	Typ.	Max.	Unit	Reference
Output rising time/falling time slew rate	tR, tF	CL = 25pF	—	25	50	ns	**Figure 37.9**
		CL = 50pF	—	50	100	ns	
		CL = 75pF	—	70	120	ns	
		CL = 100pF	—	85	150	ns	

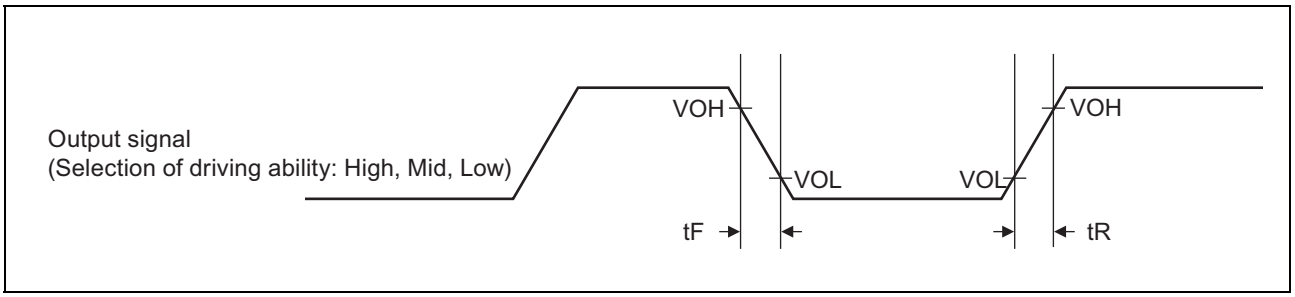


Figure 37.9 Output Signal Timing

Table 37.24 TTLVCC Power Supply Pin

Item	Symbol	Condition	Min.	Typ.	Max.	Unit	Remarks
Output rising time/ falling time slew rate	tR, tF	Selection of driving ability = High CL = 50pF	—	6	12	ns	10% to 90%
		Selection of driving ability = Mid CL = 50pF	—	15	30	ns	
		Selection of driving ability = Low CL = 50pF	—	50	100	ns	

37.3.5 Control Signal Timing

Table 37.25 Control Signals

Item	Symbol	Min.	Typ.	Max.	Unit	Reference
Reset pulse width*1	tRESW6	1.5	—	—	µs	**Figure 37.10**
Reset noise cancel width	tRESNCW	0.2	0.4	1.2	µs	
$\overline{\text{TRST}}$ noise cancel width	tTRSTNCW	0.2	0.4	1.2	µs	
NMI noise cancel width	tNC	0.2	0.4	1.2	µs	
IRQ pulse width*2	tIRQ	50	—	—	ns	
Operating mode setup time	tMDS	1	—	—	ms	**Figure 37.11**
Operating mode hold time	tMDH	1	—	—	ms	

Note 1. The reset pulse width must be equal to or more than the minimum tRESW6 value.
If the reset pulse width is less than the minimum value of the reset noise cancel width, the reset cannot be accepted.

Note 2. In case noise removal is disabled by DNF.

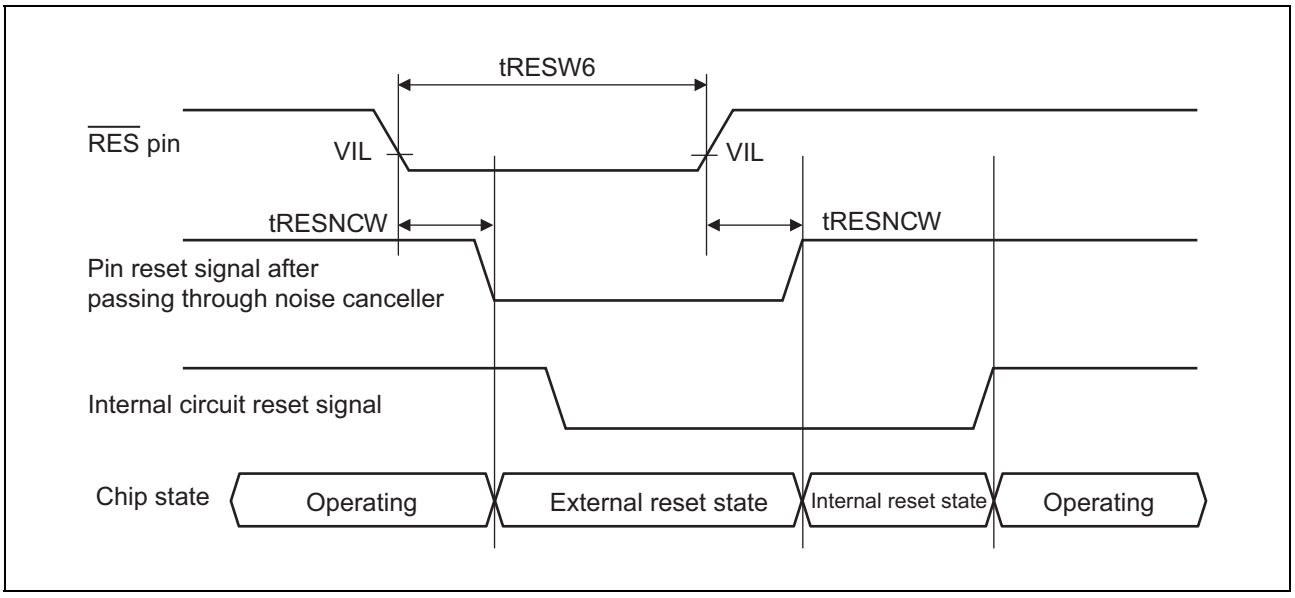


Figure 37.10 Reset Timing

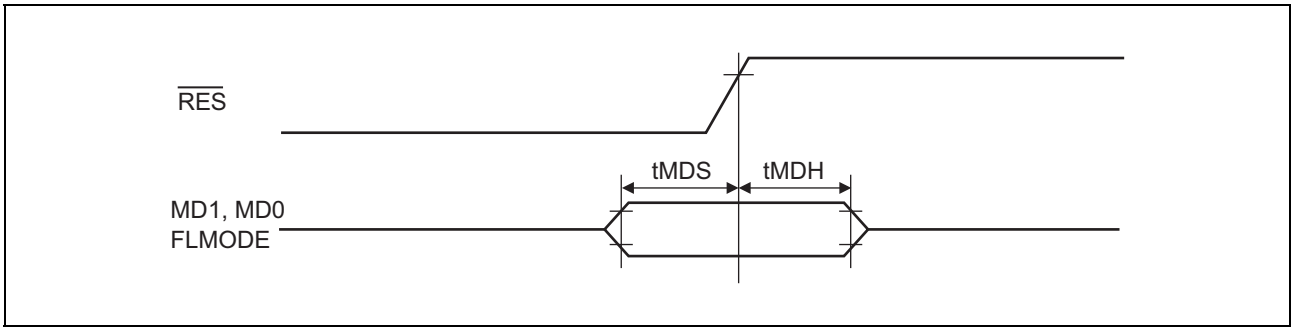


Figure 37.11 Control Signal Timing

37.3.6 CSIH Timing

Table 37.26 CSIH Timing in Master Mode

Conditions: CL = 50 pF, selection of driving ability = High, EVCC = 4.5 V to 5.5 V, TTLVCC = 4.5 V to 5.5 V

Item	Symbol	Condition	Min.	Max.	Unit	Reference
CSIHnTSCK cycle	tKCYM		100	—	ns	**Figure 37.12**
CSIHnTSCK high-level width	tKWHM		(tKCYM/2) - 12	—	ns	
CSIHnTSCK low-level width	tKWLM		(tKCYM/2) - 12	—	ns	
CSIHnTSI setup time (for CSIHnTSCK)	tSSIM		18	—	ns	
CSIHnTSI hold time (for CSIHnTSCK)	tHSIM		10	—	ns	
CSIHnTSO output delay time (for CSIHnTSCK)	tDSOM		—	10	ns	
CSIHnTSO Output hold time (for CSIHnTSCK)	tHSOM		tKWHM - 10		ns	
CSIHnTRY setup time (for CSIHnTSCK)	tSRYI	HSE = 1	(2 × tPAck) + 30	—	ns	**Figure 37.13**
CSIHnTCSSx inactive level width	tWSCSB	*1	(CSidle + 0.5) × tKCYM-20	—	ns	**Figure 37.14**
		Other than above	CSidle × tKCYM - 20	—	ns	
CSIHnTCSSx setup time (for CSIHnTSCK)	tSSCSB0	DAP = 0	CSsetup × tKCYM - 10	—	ns	
	tSSCSB1	DAP = 1	(CSsetup + 0.5) × tKCYM - 10	—	ns	
CSIHnTCSSx hold time (for CSIHnTSCK)	tHSCSB0	SIT = 0	CShold × tKCYM - 10	—	ns	**Figure 37.15**
	tHSCSB1	SIT = 1	(CShold + 0.5) × tKCYM - 10	—	ns	

Note 1. When the serial clock level is changed during communication and when the idle time is set to 0.5 transmission clock periods.

Note:
- tPAck is the operating clock cycle of CSIH (80 MHz SSCG).
- n = 0 to 3, x = 0 to 5 (n = 0), x = 0 to 3 (n = 1, 2, 3)
- CSsetup: CSIHnCFGx.CSIHnSPx3-0 set value
- CShold: CSIHnCFGx.CSIHnHDx3-0 set value
- CSidle: CSIHnCFGx.CSIHnIDx [2:0] set value
- DAP: CSIHnCFGx.CSIHnDAPx bit
- SIT: CSIHnCTL1.CSIHnSIT bit

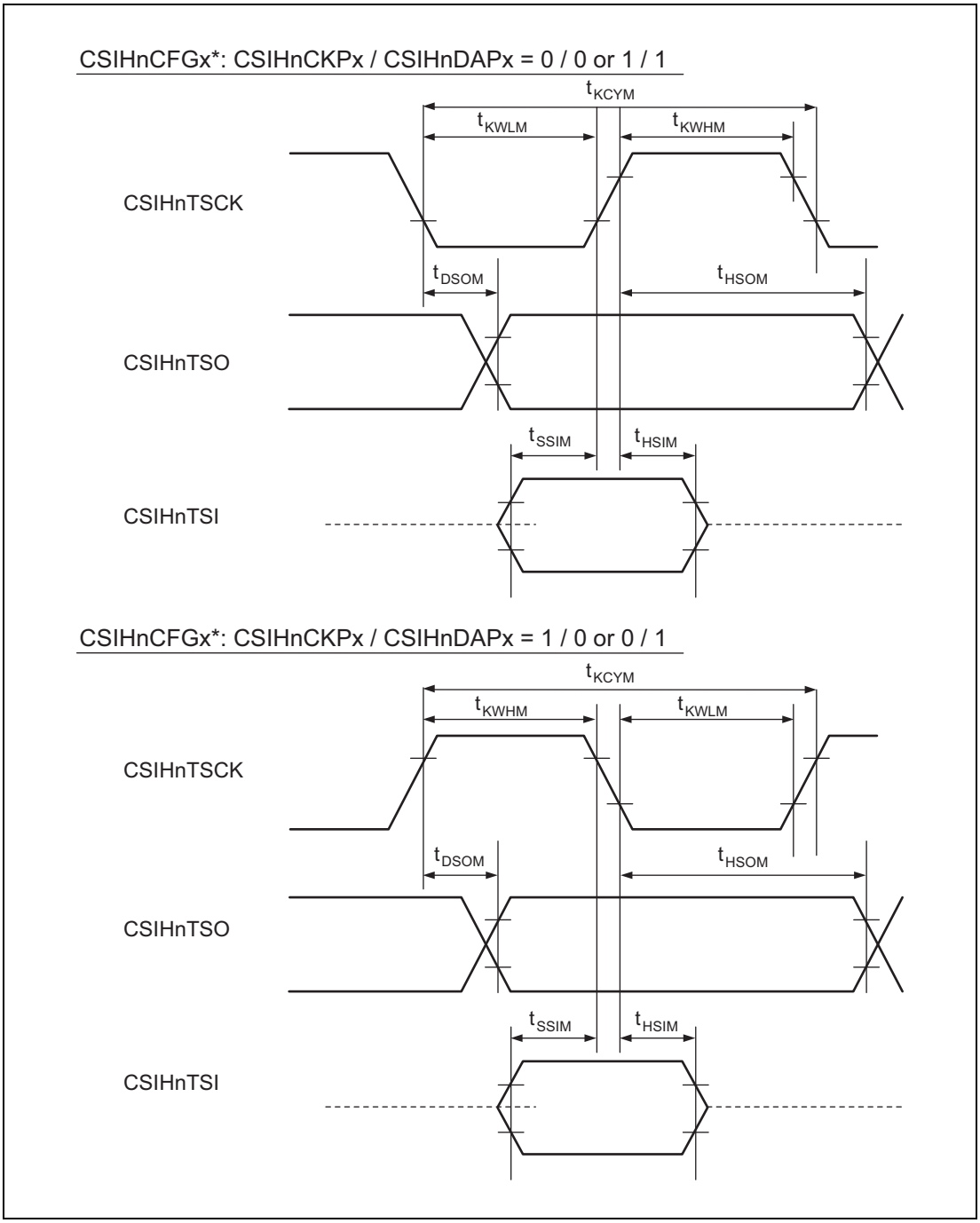


Figure 37.12 CSIH Timing (Master Mode)

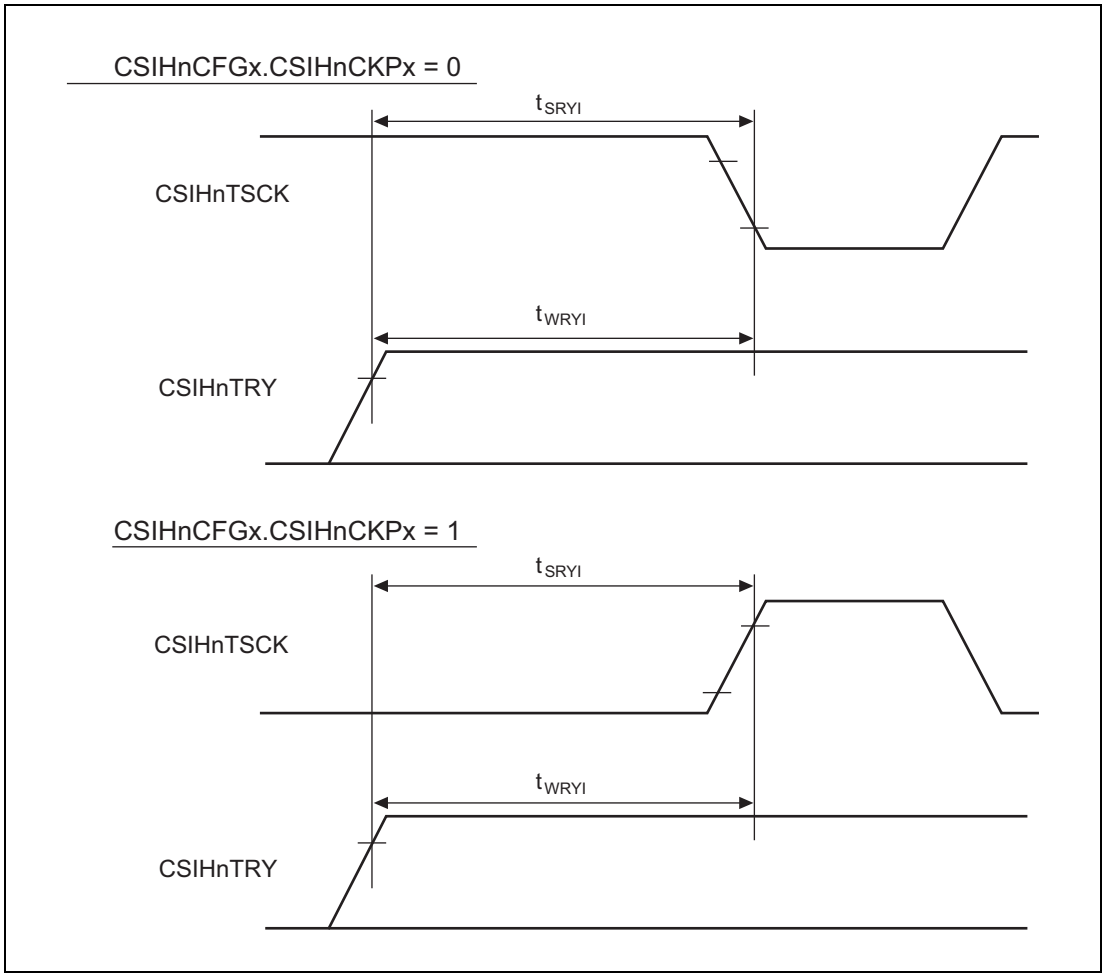


Figure 37.13 CSIH Timing (Master Mode)

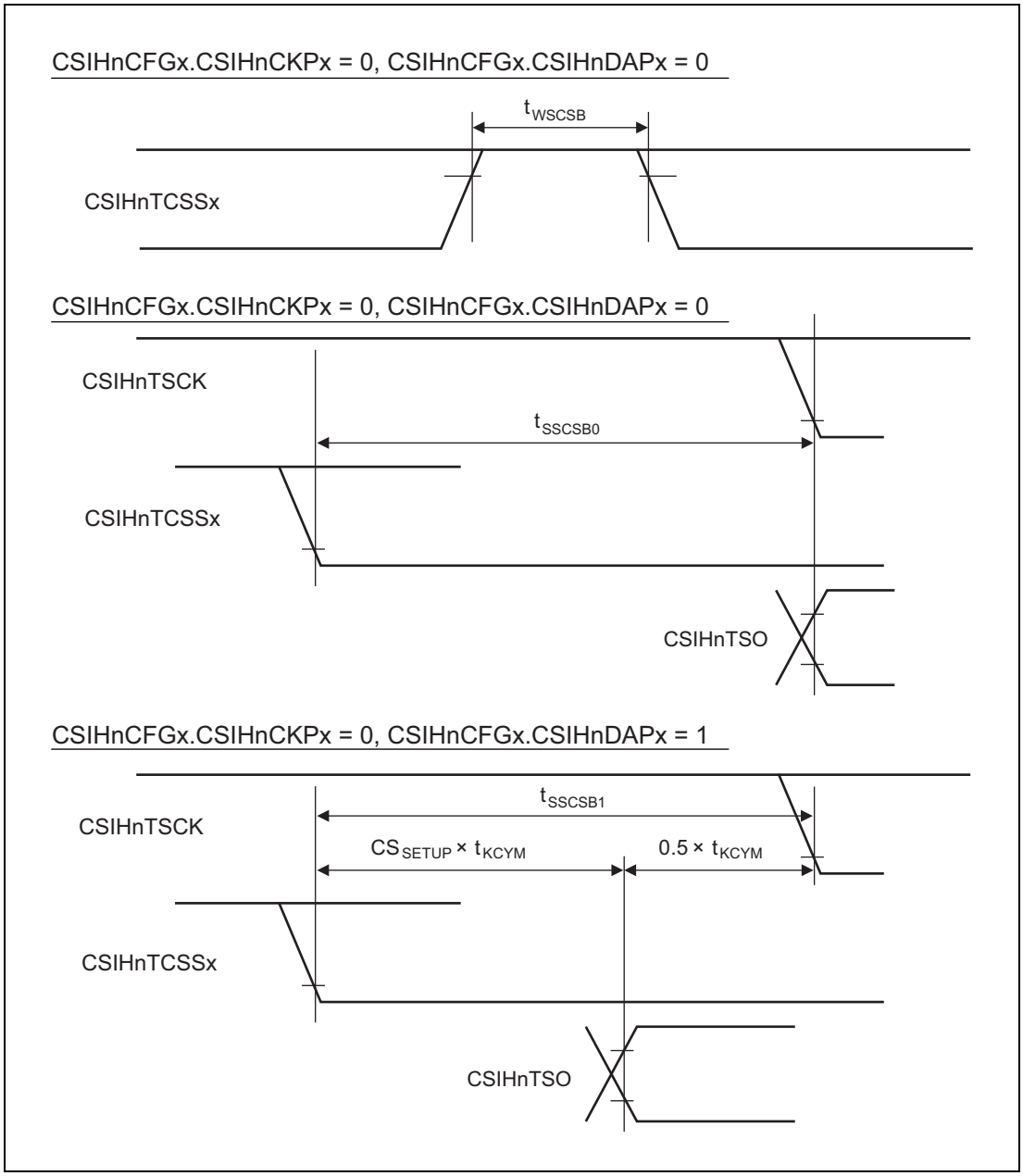


Figure 37.14 CSIH Timing (Master Mode)

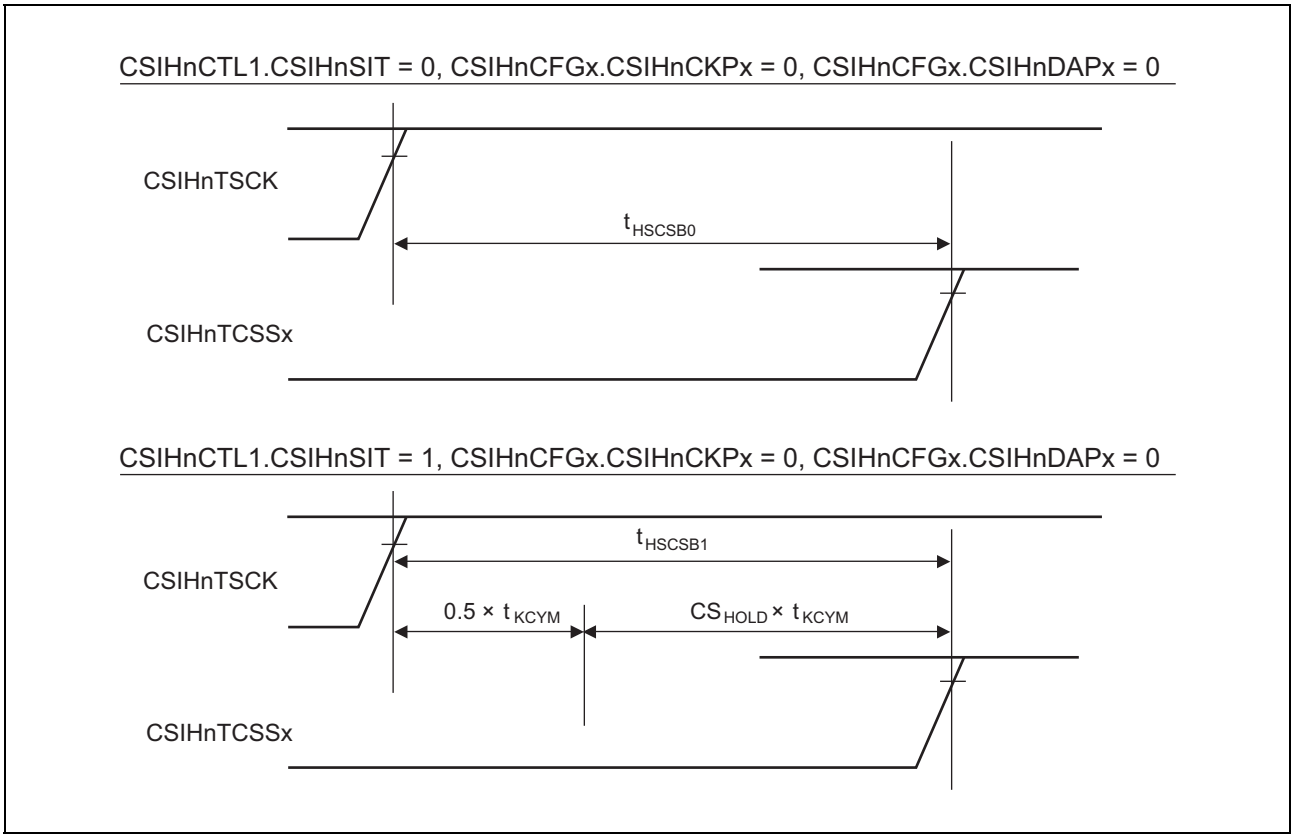


Figure 37.15 CSIH Timing (Master Mode)

Table 37.27 CSIH Timing in Slave Mode

Conditions: CL = 50 pF, selection of driving ability = High, EVCC = 4.5 V to 5.5 V, TTLVCC = 4.5 V to 5.5 V

Item	Symbol	Condition	Min.	Max.	Unit	Reference
CSIHnTSCK cycle	tKCYS		250	—	ns	**Figure 37.16**
CSIHnTSCK high-level width	tKWHS		(tKCYS/2) - 30	—	ns	
CSIHnTSCK low-level width	tKWLS		(tKCYS/2) - 30	—	ns	
CSIHnTSI setup time (for CSIHnTSCK)	tSSIS		15	—	ns	
CSIHnTSI hold time (for CSIHnTSCK)	tHSIS		tPAck + 15	—	ns	
CSIHnTSO output delay time (for CSIHnTSCK)	tDSOS		—	30	ns	
CSIHnTSO Output hold time (for CSIHnTSCK)	tHSOS		tKWHS		ns	
CSIHnTRY output delay time	tSRYO		—	30	ns	**Figure 37.17**
CSIHnTSO slave output release time	tREL			8 × tPAck	ns	
$\overline{\text{CSIHnTSSI}}$ setup time (for CSIHnTSCK)	tSSSIS		0.5 × tKCYS	—	ns	**Figure 37.18**
$\overline{\text{CSIHnTSSI}}$ hold time (for CSIHnTSCK)	tHSSIS		tPAck + 30	—	ns	

Note: tPAck is the operating clock cycle of CSIH (80 MHz SSCG).

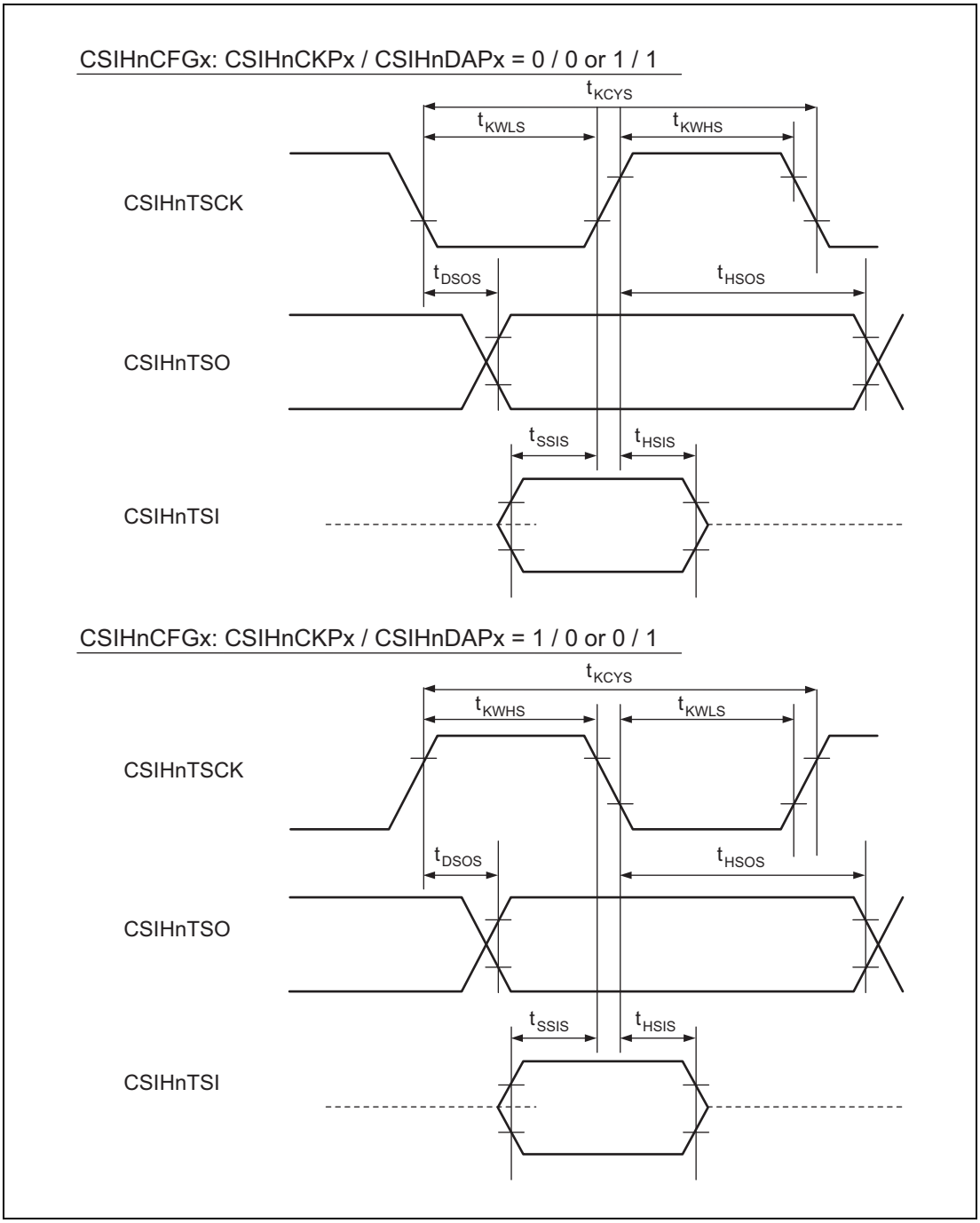


Figure 37.16 CSIH Timing (Slave Mode)

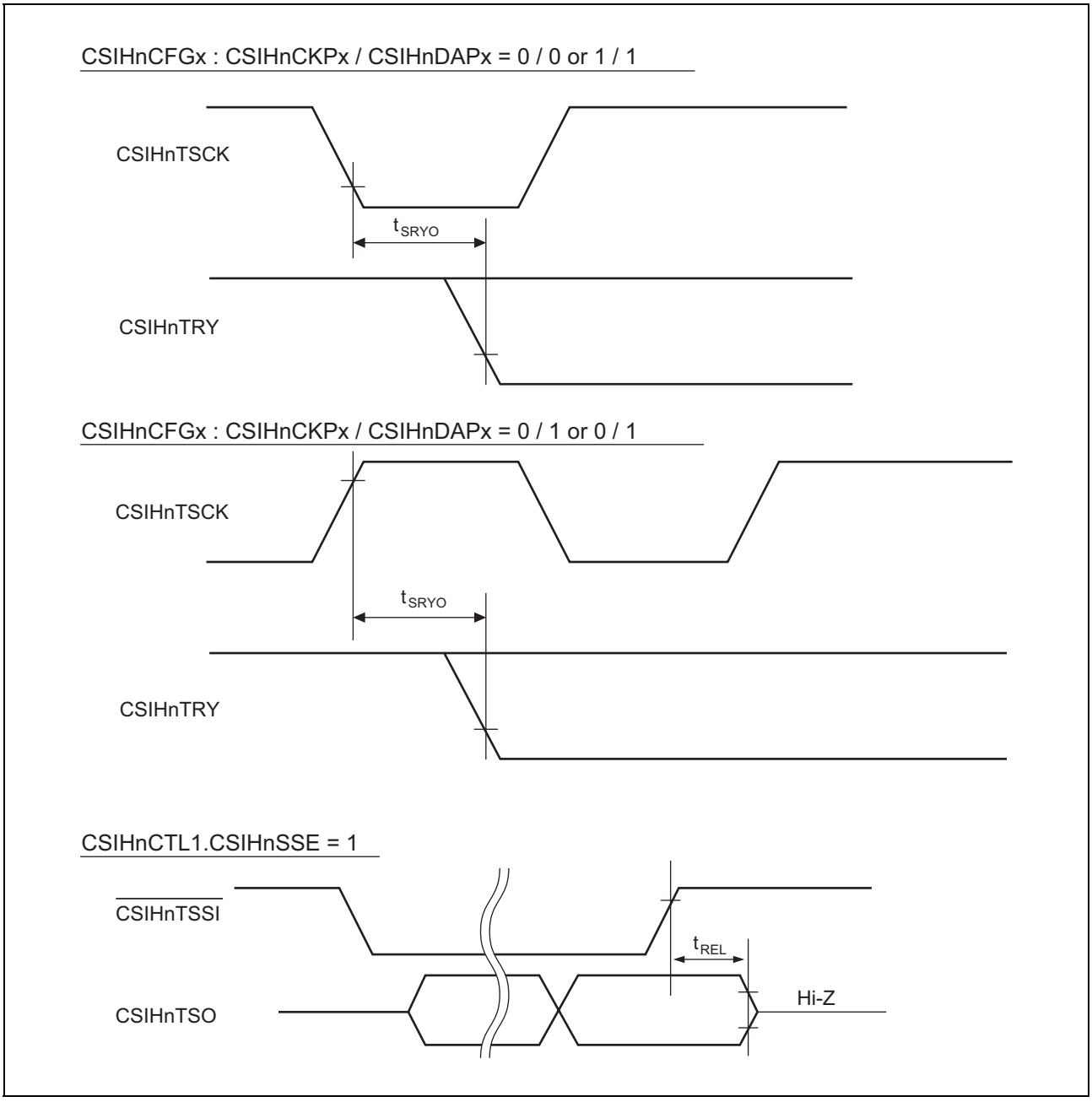


Figure 37.17 CSIH Timing (Slave Mode)

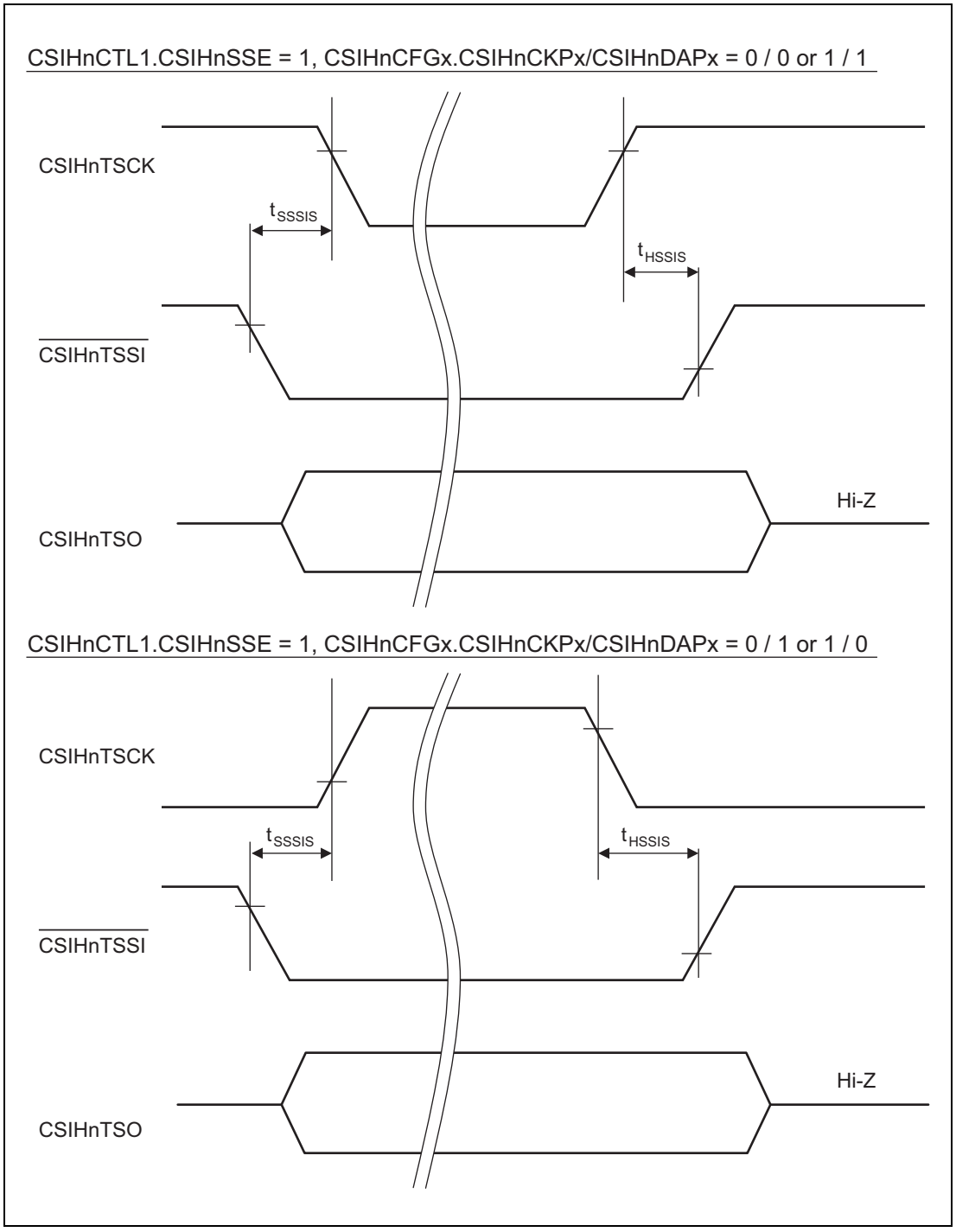


Figure 37.18 CSIH Timing (Slave Mode)

37.3.7 SCI3/FLSCI3 Timing

Table 37.28 SCI3 Timing (Master Mode)

Conditions: CL = 50 pF, selection of driving ability = High

Item	Symbol	Condition	Min.	Max.	Unit	Reference
Output clock cycle	tScyc	Asynchronous	16 × tPck	—	ns	**Figure 37.19**
		Clock synchronous	8 × tPck	—	ns	
Output clock pulse width	tSCKW		0.4 × tScyc	0.6 × tScyc	ns	
Transmit data delay time	tTXD	Clock synchronous	– 40	40	ns	**Figure 37.20**
Receive data setup time	tRXS	Clock synchronous	2 × tPck	—	ns	
Receive data hold time	tRXH	Clock synchronous	2 × tPck	—	ns	

Note: tPck is the operating clock cycle of SCI3 (40 MHz clean clock).

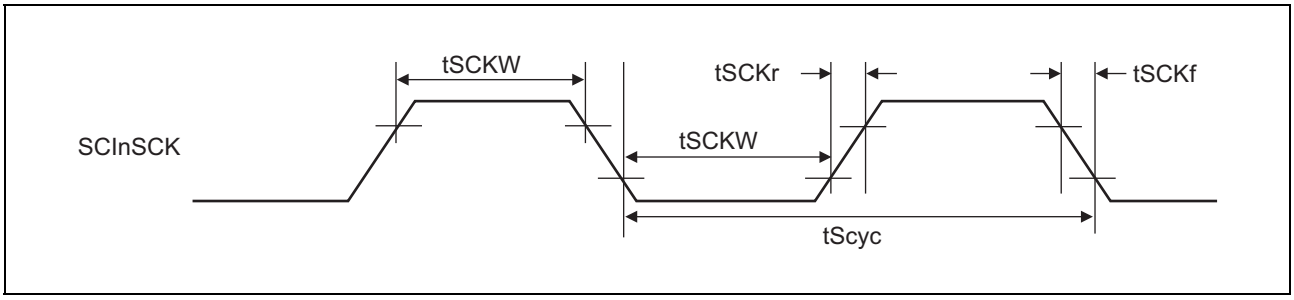


Figure 37.19 SCI3 Clock Input/Output Timing

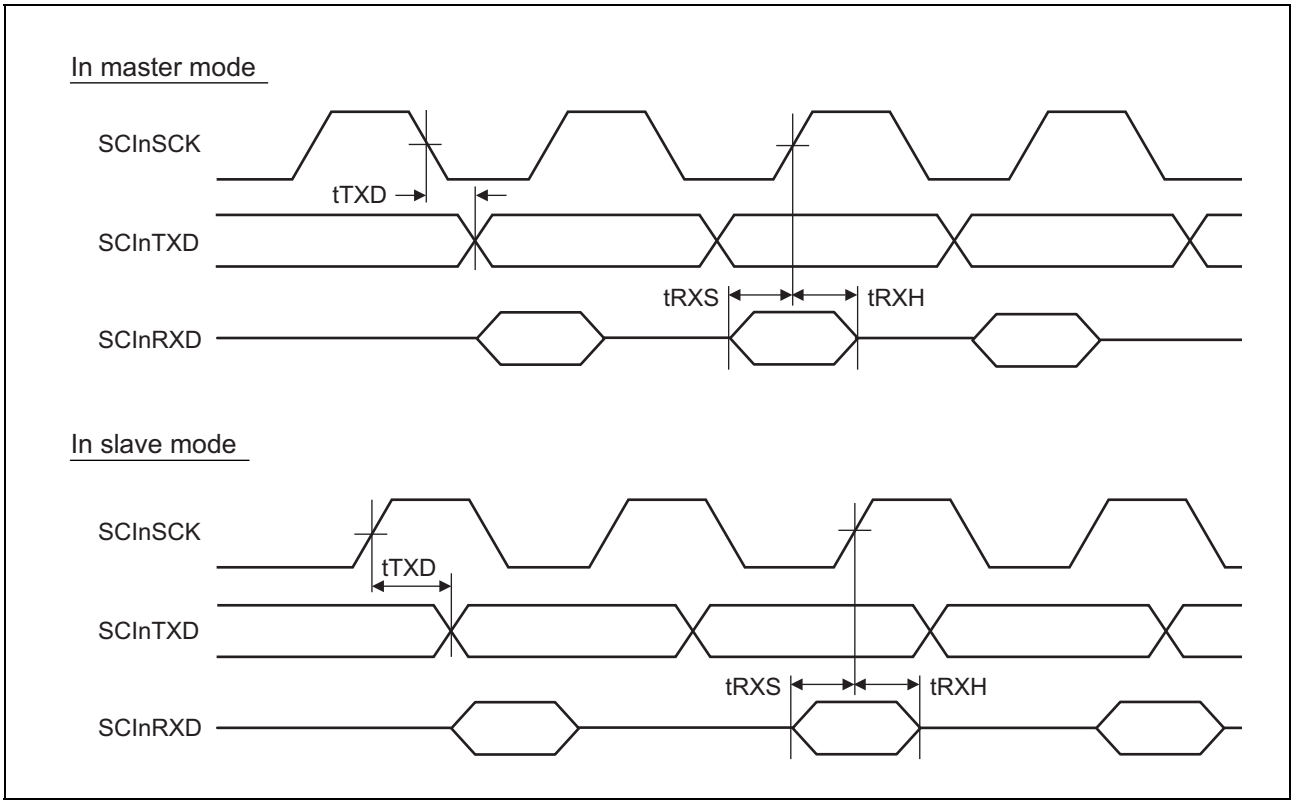


Figure 37.20 SCI3 Input/Output Timing / Clock Synchronous Mode

Table 37.29 SCI3 Timing (Slave Mode)

Conditions: CL = 50 pF, selection of driving ability = High

Item	Symbol	Min.	Max.	Unit	Reference
Input clock cycle	tScyc	8 × tPck	—	ns	**Figure 37.19**
Input clock pulse width	tSCKW	0.4 × tScyc	0.6 × tScyc	ns	
Input clock rising time	tSCKr	—	20	ns	
Input clock falling time	tSCKf	—	20	ns	
Transmit data delay time	tTXD	2 × tPck	50 + 3 × tPck*1	ns	**Figure 37.20**
Receive data setup time	tRXS	2 × tPck	—	ns	
Receive data hold time	tRXH	2 × tPck	—	ns	

Note 1. For bits other than for data 0 (1st bit) in discontinuous transfer.
Transmission of data 0 (1st bit) in discontinuous transfer starts at the same time as TDRE is set to 0.

Note: tPck represents the operating clock cycle of SCI3 (40 MHz clean clock).

37.3.8 FlexRay Timing

Table 37.30 FlexRay Timing

Conditions: CL = 15 pF, selection of driving ability = High

Item	Symbol	Condition	Min.	Max.	Unit	Reference
FRTxD transmit data rising time	tCTXr		—	2.5	ns	**Figure 37.21**
FRTxD transmit data falling time	tCTXf		—	2.5	ns	

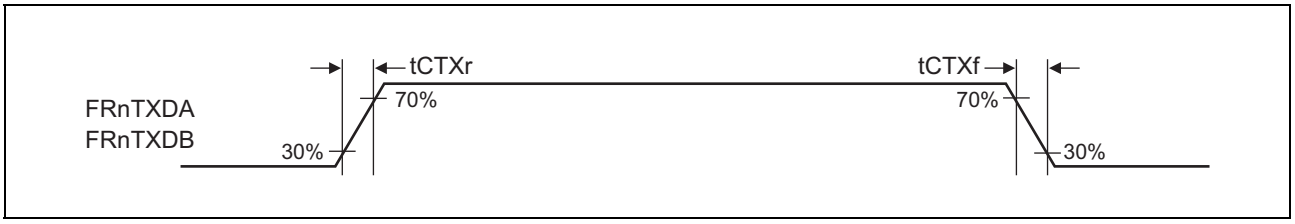


Figure 37.21 FlexRay Timing

37.3.9 RS-CAN Timing

Table 37.31 RS-CAN Timing

Conditions: CL = 50pF, selection of driving ability = High

Item	Symbol	Min.	Max.	Unit	Reference
Transmit rate			1	Mbps	**Figure 37.22**
Internal delay time	tNODE	—	100	ns	

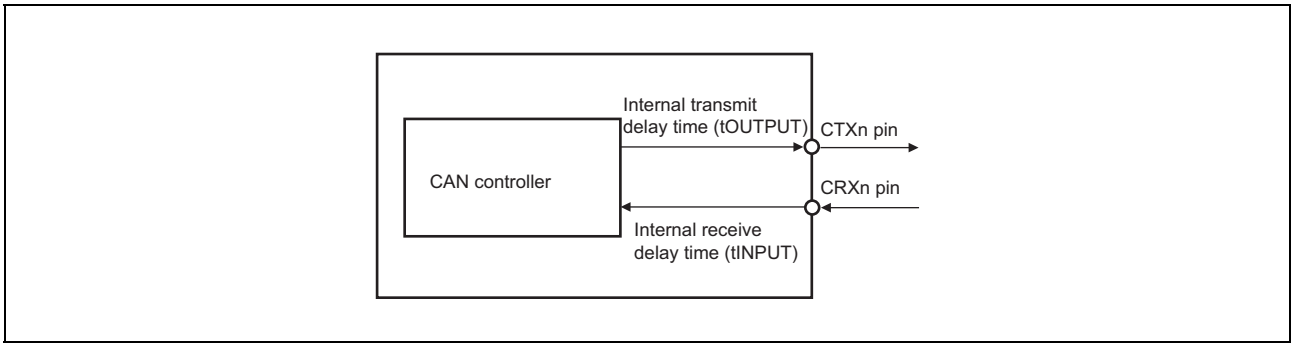


Figure 37.22 Definition of Internal Delay Time of RS-CAN
Internal delay time (tNODE) = tOUTPUT + tINPUT

37.3.10 TSG2 Timing

Table 37.32 TSG2 Timing

Item	Symbol	Condition	Min.	Max.	Unit	Reference
Input high-level width	tTIH	TAPTS00 to TAPTS02 TAPTS10 to TAPTS12 ESO2, ESO3	1.5 × tPck	—	ns	**Figure 37.23**
Input low-level width	tTIL	TAPTS00 to TAPTS02 TAPTS10 to TAPTS12 ESO2, ESO3	1.5 × tPck	—	ns	

Note: tPck is the operating clock cycle of TSG2.

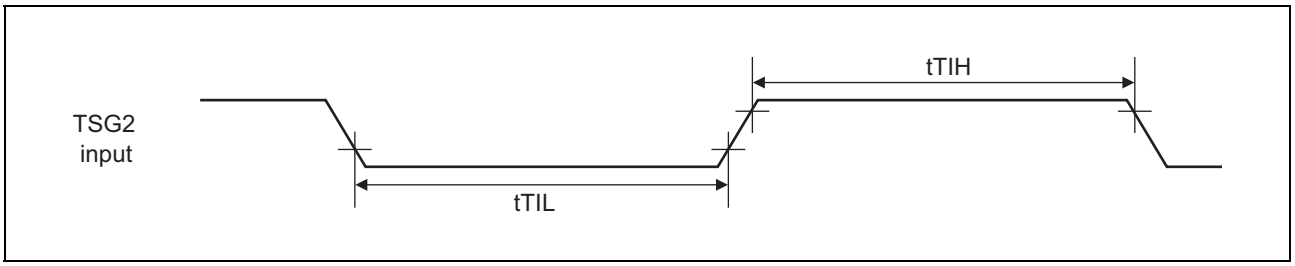


Figure 37.23 TSG2 Timing

37.3.11 RHSB Timing

Table 37.33 RHSB Timing

Conditions: LVDVCC = TTLVCC = 3.3V ± 0.3V
or LVDVCC = 3.3 ± 0.3V, TTLVCC = 5.0 ± 0.5V
CL = 50 pF

Item	Symbol	Condition	Min.	Max.	Unit	Reference
RHSBnFCL clock cycle	tFCLcyc		25	—	ns	**Figure 37.24**
RHSBnFCL high-level period	tFCLWH		0.45 × tFCLcyc	0.55 × tFCLcyc	ns	
RHSBnFCL low-level period	tFCLWL		0.45 × tFCLcyc	0.55 × tFCLcyc	ns	
RHSBnSO/RHSBnCSDx output delay time	tSOD		-0.25 × tFCLcyc	0.25 × tFCLcyc	ns	
RHSBnSI rising time	tSIr	10 to 90%	—	0.5 × tFCLcyc	ns	
RHSBnSI falling time	tSIf	10 to 90%	—	0.5 × tFCLcyc	ns	
RHSBnSI 1-bit length	tSIW		8×tFCLcyc	—	ns	

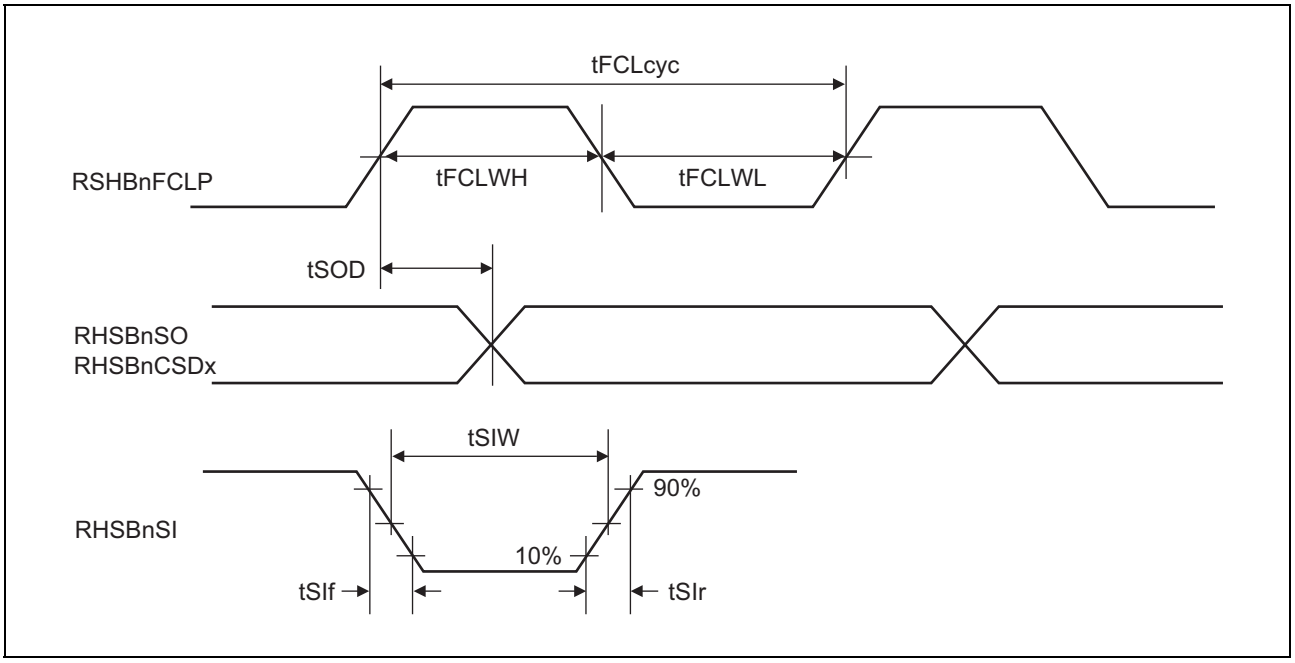


Figure 37.24 RHSB Timing

CAUTION

1. **When the following pins are not used as LVDS, leave the RHSB*N pins open. In that case, the pin load capacitance must be 0.2 pF or less.**
   [Pin pair]
   - **RHSBnFCLN/RHSBnFCLP (n = 0, 1)**
   - **RHSBnSON/RHSBnSOP (n = 0, 1)**
2. **RHSBnCSDx does not satisfy the specification if Mid or Low is selected for the driving ability of output pins. When P13_0, P13_1, P13_5, or P13_6 is used as an RHSB pin, driving ability must be set to High.**

37.3.12 JTAG/NEXUS Timing

Table 37.34 JTAG/NEXUS Timing

Conditions: Tj = -40 to 150°C, CL = 30 pF

Item	Symbol	Condition	Min.	Max.	Unit	Reference
TCK cycle time	tTCKW		40	—	ns	**Figure 37.25**
TCK high-level width	tTCKWH		16	—	ns	
TCK low-level width	tTCKWL		16	—	ns	
TMS/TDI setup time (until TCK ↑)	tTISU		12	—	ns	
TMS/TDI hold time (until TCK ↑)	tTIH		12	—	ns	
TDO output delay time (until TCK ↓)	tTDOD		—	tTCKW - 20	ns	
$\overline{\text{DRDY}}$ output delay time (until TCK ↓)	tRDYD		—	tTCKW - 20	ns	
$\overline{\text{TRST}}$ low-level width	tTRSTWL		1200	—	ns	
TCK/$\overline{\text{TRST}}$/TMS/TDI input rising time	tTIR		—	12	ns	
TCK/$\overline{\text{TRST}}$/TMS/TDI input falling time	tTIF		—	12	ns	

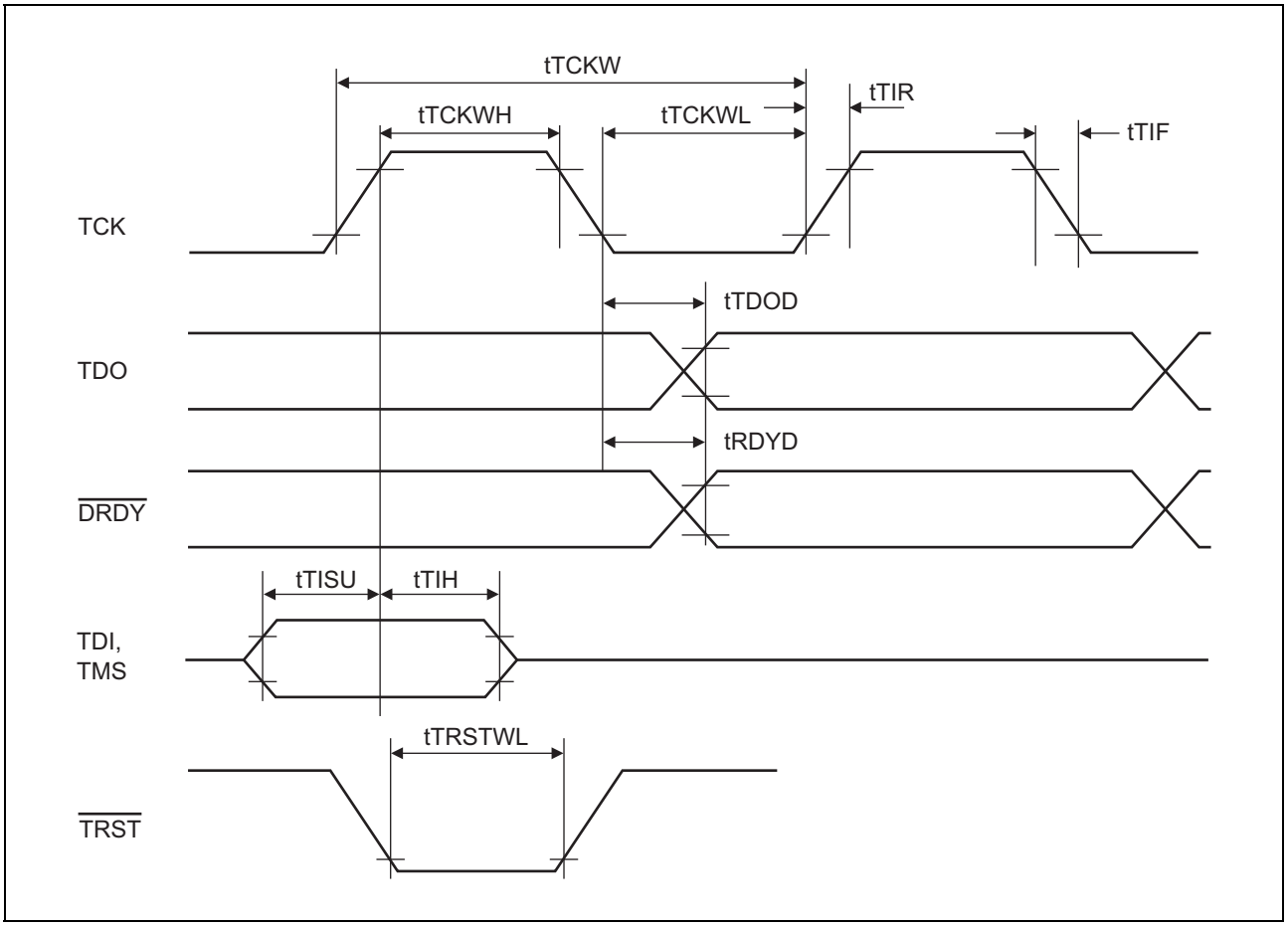


Figure 37.25 JTAG/NEXUS Timing

37.3.13 LDU 4-Wire Timing

Table 37.35 LDU 4-Wire Timing

Conditions: Tj = -40 to 150°C, CL = 30 pF

Item	Symbol	Condition	Min.	Max.	Unit	Reference
LPDCLK cycle time	tLPDCKW		25	—	ns	**Figure 37.26**
LPDCLK high-level width	tLPDCKWH		4.5	—	ns	
LPDCLK low-level width	tLPDCKWL		4.5	—	ns	
LPDCLK input rising time	tLPDCKR		—	8	ns	
LPDCLK input falling time	tLPDCKF		—	8	ns	
LPDI setup time (until LPDCLK ↑)	tLPDSU		2	—	ns	
LPDI hold time (until LPDCLK ↑)	tLPDH		2	—	ns	
LPDCLKO cycle time	tLPDCKOW		25	—	ns	
LPDCLKO high-level width	tLPDCKOWH		4.5	—	ns	
LPDCLKO low-level width	tLPDCKOWL		4.5	—	ns	
LPDCLKO rising time	tLPDCKOR		—	8	ns	
LPDCLKO falling time	tLPDCKOF		—	8	ns	
LPDO output delay (until LPDCLKO ↑)	tLPDOD		0	12	ns	

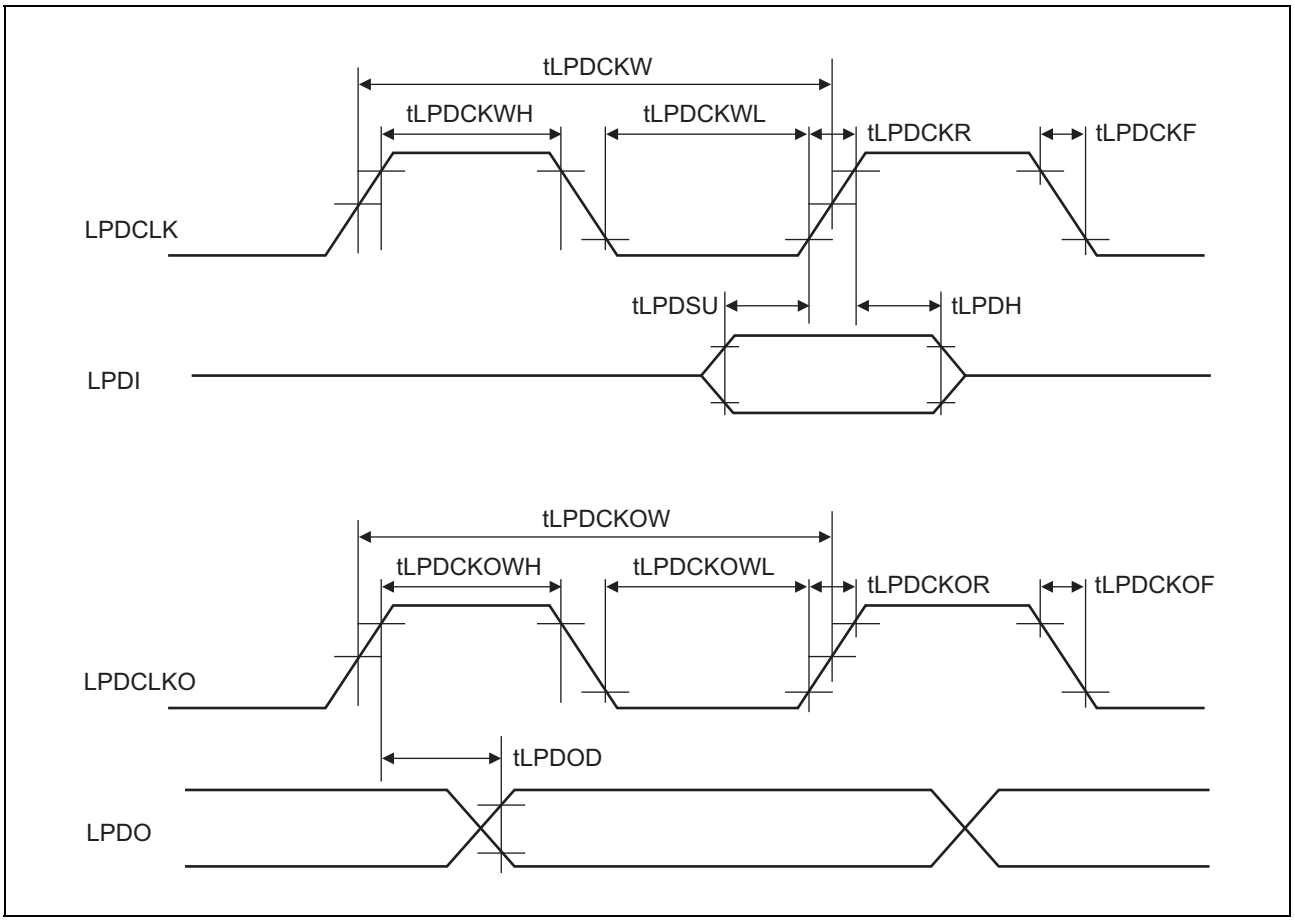


Figure 37.26 LDU 4-Wire Timing

37.3.14 AUD RAM Monitor

Table 37.36 AUD RAM Monitor Timing

Conditions: Tj = -40 to 125 °C, CL = 30 pF

Item	Symbol	Min.	Max.	Unit	Reference
AUDCK cycle time (monitor mode)	tAUCKMcyc	50	—	ns	**Figure 37.27**
AUDCK high-level width (monitor mode)	tAUCKMH	0.4		tAUCKMcyc	
AUDCK low-level width (monitor mode)	tAUCKML	0.4		tAUCKMcyc	
$\overline{\text{AUDRST}}$ setup time (monitor mode)	tAMRSTMS	30		ns	
$\overline{\text{AUDRST}}$ pulse width (monitor mode)	tAURSTMW	5		tAUCKMcyc	
Monitor data output delay time	tAUDTMD	—	35	ns	
Monitor data input setup time	tAUDTMS	15		ns	
Monitor data input hold time	tAUDTMH	5		ns	
$\overline{\text{AUDSYNC}}$ input setup time	tAUDSYS	15		ns	
$\overline{\text{AUDSYNC}}$ input hold time	tAUDSYH	5		ns	

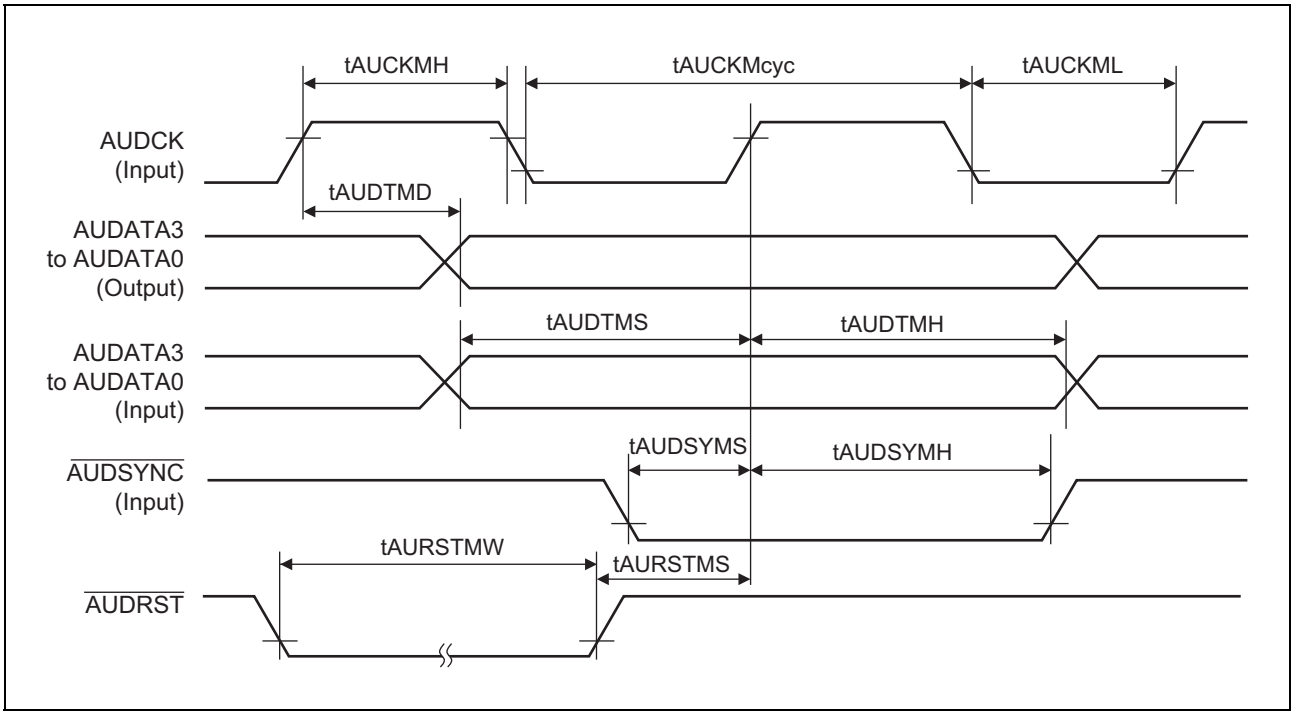


Figure 37.27 AUD RAM Monitor Timing

37.4 A/D Converter Characteristics

Table 37.37 ADC Converter Characteristics

Item		Symbol	Min.	Typ.	Max.	Unit	Reference
Digital resolution		—	—	12	—	bit	
Voltage resolution*1		—	1.10	1.22	1.34	mV	
A/D conversion time (f_{OP} = 20 MHz, 40 MHz)		—	—	1.0	—	µs	
Non-linearity error		—	—	—	±4.0	LSB	
Offset error		—	—	—	±7.5	LSB	
Full-scale error		—	—	—	±7.5	LSB	
Quantization error		—	—	—	±0.5	LSB	
Absolute error		—	—	—	±8.0	LSB	
Self-diagnosis absolute error	A/D conversion circuit self-diagnosis is in use	—	—	—	±8.0	LSB	
	Pin-level self-diagnosis is in use	—	—	—	±40	LSB	
Analog input capacitance	Waiting	—	—		10	pF	
	During sampling	—	—		20	pF	
Allowable analog signal source impedance		—	—	—	3	kΩ	
T&H hold time		—	—	—	10	µs	
Input Voltage Range	When T&H is not used	—	0		A0VREFH, A1VREFH	V	
	When T&H is used	—	0.2		A0VREFH, A1VREFH – 0.2	V	

Note 1. When AnVREFH (n = 0, 1) = 4.5 V, the resolution is 1.10 mV. When AnVREFH (n = 0, 1) = 5.5 V, the resolution is 1.34 mV.

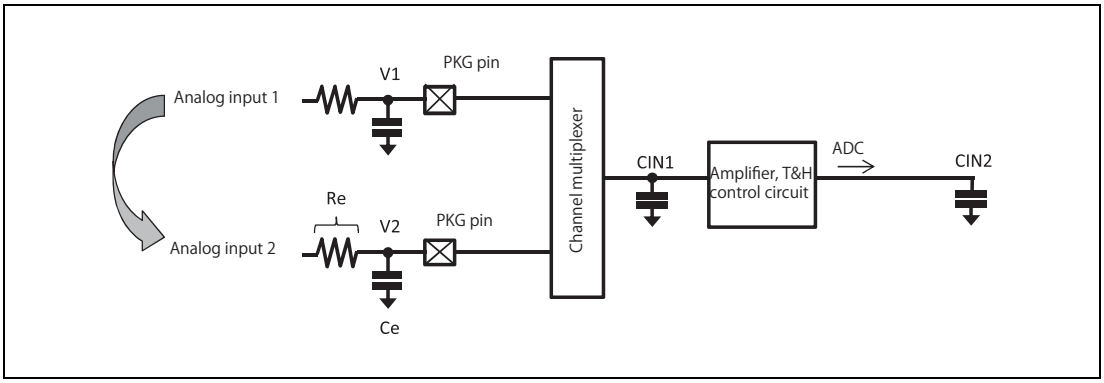
CAUTION

A current being injected to analog input pins while pin-level self-diagnosis is in use does not guarantee the conversion accuracy for the diagnostic voltage of the channels for these pins.

- Errors in the External Circuit of the A/D Convertor
  A formula for errors in sampled values due to the external circuit of the A/D converter is given below.
  These errors will depend on the input circuit and conversion cycle. The formula given below for the errors is simplified for the calculation of sampling errors based on internal stray capacitance, amplifier offset, resistance of the signal source, and conversion cycle. This formula can also be used to calculate the effects of the signal source resistance and conversion cycle on these errors. The formula gives the error of analog input 2 as shown in the figure below when A/D conversion is performed in the order analog input 1 then 2.

$$\text{Conversion error(LSB)} = \left[\left(\frac{|V2\text{-}V1| \times CIN1}{Ce + CIN1} + \frac{|Vvfaerr| \times CIN2}{Ce + CIN2} \right) \times \frac{1}{1\text{-}e^{-T1/(Re \times Ce)}} + \left(\frac{1}{T1} \times C1 \times V3 \times Re \right) \right] \times \frac{4096}{Vavrefh}$$

Item	Symbol	Reference	Unit
Common capacitance of the final stage of channel multiplexer	CIN1	1.4	pF
Common capacitance of the final state of the amplifier and T&H control circuit	CIN2	10	pF
External capacitor on analog input pin	Ce	Depends on user board	μF
Signal source impedance	Re		KΩ
Conversion cycle of conversion pins	T1		ms
AnVREFH voltage (n = 0, 1)	Vavrefh		V
Potential difference between V1 and V2	V2-V1	5	V
Offset voltage of amplifier and T&H control circuit	Vvfaerr	50	mV
Parasitic capacitance of the last stage of channel multiplexer	C1	2	pF
|AnVCC voltage /2.5 – measured pin voltage (V2)| (n = 0, 1)	V3	Depends on user board	V



- Values for conversion error calculated by using this formula do not include error (absolute error, etc.) specified in the A/D converter characteristics.
- This formula is a desktop formula and theoretical. When the signal source has an extremely high resistance or when the conversion cycle is too short, calculated and measured values may differ. Actual error depends on the capacitor, resistor, capacitance and resistance of board wiring, so please evaluate and verify the error on the user board is no greater than the value produced by this formula (Re < 1.5 MΩ and T1 ≥ 10μs, or 1.5 MΩ ≤ Re ≤ 2 MΩ and T1 ≥ 512 μs).

Table 37.38 ΔΣADC Conversion Characteristics (1/2)

Example 1
DSADCmCCR.DSDFTYP = 0_H Fs = 100 ksps
DSADCmFCR.ORT = 1_H BW = 30 kHz
DSADCmFCR.TPVSL[2:0] = 1_H GIN = × 2

Item	Symbol	Condition	Min.	Typ.	Max.	Unit
Valid bit	ENOB	—	13	—	—	bit
Peak SNDR	SNDR	—	80	—	—	dB
Shutoff frequency	BW	fos = 8 MHz	—	—	30	kHz
Initial delay	—	—	—	—	65	µs
Group delay	—	Fin = up to 10 kHz	—	—	15	µs
Gain error	Gerr	DC input	–1	—	1	%

Example 2
DSADCmCCR.DSDFTYP = 0_H Fs = 200 ksps
DSADCmFCR.ORT = 0_H BW = 30 kHz
DSADCmFCR.TPVSL[2:0] = 1_H GIN = × 2

Item	Symbol	Condition	Min.	Typ.	Max.	Unit
Valid bit	ENOB	—	13	—	—	bit
Peak SNDR	SNDR	—	80	—	—	dB
Shutoff frequency	BW	fos = 8 MHz	—	—	30	kHz
Initial delay	—	—	—	—	65	µs
Group delay	—	Fin = up to 10 kHz	—	—	15	µs
Gain error	Gerr	DC input	–1	—	1	%

Example 3
DSADCmCCR.DSDFTYP = 0_H Fs = 200 ksps
DSADCmFCR.ORT = 0_H BW = 60 kHz
DSADCmFCR.TPVSL[2:0] = 2_H GIN = × 2

Item	Symbol	Condition	Min.	Typ.	Max.	Unit
Valid bit	ENOB	—	13	—	—	bit
Peak SNDR	SNDR	—	80	—	—	dB
Shutoff frequency	BW	fos = 8 MHz	—	—	60	kHz
Initial delay	—	—	—	—	65	µs
Group delay	—	Fin = up to 10 kHz	—	—	10	µs
Gain error	Gerr	DC input	–1	—	1	%

Example 4
DSADCmCCR.DSDFTYP = 1_H Fs = 1.6 Msps
DSADCmFCR.ORT = 0_H BW = 200 kHz
DSADCmFCR.TPVSL[2:0] = 0_H GIN = × 2

Item	Symbol	Condition	Min.	Typ.	Max.	Unit
Valid bit	ENOB	—	8	—	—	bit
Peak SNDR	SNDR	—	50	—	—	dB
Shutoff frequency	BW	fos = 8 MHz	—	—	200	kHz
Initial delay	—	—	—	—	4.588	µs
Group delay	—	Fin = up to 10 kHz	—	—	2.025	µs
Gain error	Gerr	DC input	–1	—	1	%

Table 37.38 ΔΣADC Conversion Characteristics (2/2)

Common

Item	Symbol	Condition	Min.	Typ.	Max.	Unit
Input conversion offset error	Vos	—	–20	—	20	mV
Input impedance	—	GIN = ×1	200	—	—	kΩ
	—	GIN = ×2	100	—	—	kΩ
	—	GIN = ×4	50	—	—	kΩ
	—	GIN = ×8	25	—	—	kΩ

Table 37.39 ΔΣ-A/D Input Voltage Specifications

Item	Symbol	conditions	Min.	Typ.	Max.	Unit	Reference
Input voltage specification with reference single-end ADSVREFH/2	Vain	GIN = ×1	ADSVREFL	—	ADSVREFH	V	
		GIN = ×2	ADSVREFL	—	ADSVREFH	V	
		GIN = ×4	ADSVREFH×(1/4)	—	ADSVREFH×(3/4)	V	
		GIN = ×8	ADSVREFH×(3/8)	—	ADSVREFH×(5/8)	V	
Input voltage specification with reference single-end ADSVREFL	Vain	GIN = ×1	ADSVREFL	—	ADSVREFH	V	
		GIN = ×2	ADSVREFL	—	ADSVREFH×(1/2)	V	
		GIN = ×4	ADSVREFL	—	ADSVREFH×(1/4)	V	
		GIN = ×8	ADSVREFL	—	ADSVREFH×(1/8)	V	
Input voltage specification with differential input	Vain	GIN = ×1	-ADSVREFH	—	ADSVREFH	V	
		GIN = ×2	-ADSVREFH×(1/2)	—	ADSVREFH×(1/2)	V	
		GIN = ×4	-ADSVREFH×(1/4)	—	ADSVREFH×(1/4)	V	
		GIN = ×8	-ADSVREFH×(1/8)	—	ADSVREFH×(1/8)	V	

37.5 Code Flash Characteristics

Table 37.40 Code Flash Basic Characteristics

Item	Symbol	Conditions	Min.	Typ.	Max.	Unit
Programming endurance*1	CWRT	Retained for 20 years *2	1000	—	—	Times
Temperature range of programming	TPRG	Tj	-40	—	+150	℃
Temperature range of reading	TREAD	Tj	-40	—	+150	℃

Note 1. Programming endurance is defined as the number of times each block is erased. Where programming endurance is n times (n = 1000 in this case), each block is erasable n times. For example, given a memory device that has 32-Kbyte erasure blocks, programming in the address range of each 256-byte programming block (128 programming operations, one for each block) in an erasure block and then erasing the block counts as one time in terms of programming endurance. However, programming of a given address range more than once after erasure is not possible (overwriting after programming is prohibited).

Note 2. The retained period when the average Ta is 85°C. This retained period is from when the erasure of the code flash memory has been normally completed.

Table 37.41 Code Flash Programming Characteristics

Conditions: EVCC = 4.5 V to 5.5 V, TTLVCC = 4.5 V to 5.5 V or 3.0 V to 3.6 V, A0VCC, A1VCC = 4.5 V to 5.5 V,
A0VREFH = 4.5 V to A0VCC, A1VREFH = 4.5 V to A1VCC, ADSVCC = 4.5 V to 5.5 V,
ADSVREFH = 4.5 V to ADSVCC, PLLVCC = 3.0 V to 3.6 V, SYSVCC = 3.0 V to 3.6 V,
VCC = 3.0 V to 3.6 V, LVDVCC = 3.0 V to 3.6 V, VDD = 1.15 V to 1.35 V,
VSS = PLLVSS = A0VSS = A1VSS = ADSVSS = ADSVREFL = LVDVSS = 0 V
Tj = -40°C to 150°C

Item	Condition	Block size	Min.	Typ.	Max.	Unit
Programming time	Programming endurance < 100 times	256 B	—	0.4*1	6*1	ms
		32KB	—	80	360	ms
	Programming endurance ≥ 100 times	256 B	—	0.5*1	7.2*1	ms
		32KB	—	96	432	ms
Erasing time	Programming endurance < 100 times	8 KB	—	39	120	ms
		32 KB	—	141	480	ms
	Programming endurance ≥ 100 times	8 KB	—	47	144	ms
		32 KB	—	169	576	ms

Note 1. Values are only for processing by hardware. Software overhead is not taken into account.

37.6 Data Flash Characteristics

Table 37.42 Data Flash Basic Characteristics

Item	Symbol	Condition	Min.	Typ.	Max.	Unit
Programming endurance*1	CWRT	Retained for 20 years*2	125000	—	—	Times
		Retained for 3 years*2	250000	—	—	Times
Temperature range of programming	TPRG	Tj	-40	—	+150	°C
Temperature range of reading	TREAD	Tj	-40	—	+150	°C

Note 1. Programming endurance is defined as the number of times each block is erased. Where programming endurance is n times (n = 125000 in this case), each block is erasable n times. For example, given a memory device that has 64-byte erasure blocks, programming in the address range of each 4-byte programming block (16 programming operations, one for each block) in an erasure block and then erasing the block counts as one time in terms of programming endurance. However, programming of a given address range more than once after erasure is not possible (overwriting after programming is prohibited).

Note 2. This is the case when the average Ta is 85°C. This retained period is from when the erasure of the data flash memory has been normally completed.

Table 37.43 Data Flash Programming Characteristics

Conditions: EVCC = 4.5 V to 5.5 V, TTLVCC = 4.5 V to 5.5 V or 3.0 V to 3.6 V, A0VCC,A1VCC = 4.5 V to 5.5 V,
A0VREFH = 4.5 V to A0VCC, A1VREFH = 4.5 V to A1VCC, ADSVCC = 4.5 V to 5.5 V,
ADSVREFH = 4.5 V to ADSVCC, PLLVCC = 3.0 V to 3.6 V, SYSVCC = 3.0 V to 3.6 V,
VCC = 3.0 V to 3.6 V, LVDVCC = 3.0 V to 3.6 V, VDD = 1.15 V to 1.35 V,
VSS = PLLVSS = A0VSS = A1VSS = ADSVSS = ADSVREFL = LVDVSS = 0 V
Tj = -40°C to 150°C

Item	Block size	Min.	Typ.	Max.	Unit
Programming time	4 B	—	0.16*1	1.7*1	ms
Erasing time	64 B	—	1.7*1	10*1	ms
Blank check time	4 B	—	—	30*1	µs
	64 B	—	—	100*1	µs

Note 1. Values are only for processing by hardware. Software overhead is not taken into account.

37.7 Thermal Characteristics

37.7.1 Parameters

Table 37.44 Thermal Resistance of RH850/E1M-S

Package	Parameter	Estimate	Unit	Note
BGA1919-304	Θja	18	°C/W	Conforming to JESD51-9 (4 layers)
	Ψjb	13	°C/W	Conforming to JESD51-9 (4 layers)
	4LΘja	27	°C/W	L board (4 layers) (no thermal vias)
	4LΨjb	19	°C/W	L board (4 layers) (no thermal vias)
	4LTb_inc	8	°C/W	L board (4 layers) (no thermal vias)
	4LΨjt	1	°C/W	L board (4 layers) (no thermal vias)
BGA1717-252	Θja	19	°C/W	Conforming to JESD51-9 (4 layers)
	Ψjb	13	°C/W	Conforming to JESD51-9 (4 layers)
	4LΘja	28	°C/W	L board (4 layers) (no thermal vias)
	4LΨjb	19	°C/W	L board (4 layers) (no thermal vias)
	4LTb_inc	9	°C/W	L board (4 layers) (no thermal vias)
	4LΨjt	1	°C/W	L board (4 layers) (no thermal vias)

37.7.2 Assumed Board

Table 37.45 JESD51-9 Compliant Board (4 layers)

	Board Size (mm)		Area (mm2)
	X	Y	
Board	101.5	114.5	11621.75
Remaining copper rates		Thickness of conductors	
50-95-95-50%		70-35-35-70µm	

Table 37.46 L Board (4 layers)

	Board Size (mm)		Area (mm2)
	X	Y	
L Board	90	160	14400
Remaining copper rates		Thickness of conductors	
30-80-80-30%		35-35-35-35µm	

Appendix List of Registers

This section describes information of on-chip I/O registers of this LSI as follows.

- Registers are listed in ascending order of address (OS address).
- Do not access reserved addresses that are not listed below.

In the case of 16-bit or 32-bit addresses, LSB-side addresses are described on the assumption of little endian.

(1/201)

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
FLXA0	FlexRay Operation Control Register	FLXA0FROC	0000 0000$_H$	32	1002 0004$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Operation Status Register	FLXA0FROS	0000 0000$_H$	32	1002 000C$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Lock Register	FLXA0FRLCK	0000 0000$_H$	32	1002 001C$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay error Interrupt Register	FLXA0FREIR	0000 0000$_H$	32	1002 0020$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Status Interrupt Register	FLXA0FRSIR	0000 0000$_H$	32	1002 0024$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Error Interrupt Output Selection Register	FLXA0FREILS	0000 0000$_H$	32	1002 0028$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Status Interrupt output selection Register	FLXA0FRSILS	0303 FFFF$_H$	32	1002 002C$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Error Interrupt Enable Register	FLXA0FREIES	0000 0000$_H$	32	1002 0030$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Error Interrupt Disable Register	FLXA0FREIER	0000 0000$_H$	32	1002 0034$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Status Interrupt Enable Register	FLXA0FRSIES	0000 0000$_H$	32	1002 0038$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Status Interrupt Disable Register	FLXA0FRSIER	0000 0000$_H$	32	1002 003C$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Interrupt Output Enable Register	FLXA0FRILE	0000 0000$_H$	32	1002 0040$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Timer 0 Setting Register	FLXA0FRT0C	0000 0000$_H$	32	1002 0044$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Timer 1 Setting Register	FLXA0FRT1C	0002 0000$_H$	32	1002 0048$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Stop Watch Register 1	FLXA0FRSTPW1	0000 0000$_H$	32	1002 004C$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Stop Watch Register 2	FLXA0FRSTPW2	0000 0000$_H$	32	1002 0050$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay SUC Setting Register 1	FLXA0FRSUCC1	0C40 1080$_H$	32	1002 0080$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay SUC Setting Register 2	FLXA0FRSUCC2	0100 0504$_H$	32	1002 0084$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay SUC Setting Register 3	FLXA0FRSUCC3	0000 0011$_H$	32	1002 0088$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay NEM Setting Register	FLXA0FRNEMC	0000 0000$_H$	32	1002 008C$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay PRT Setting Register 1	FLXA0FRPRTC1	084C 0633$_H$	32	1002 0090$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay PRT Setting Register 2	FLXA0FRPRTC2	0F2D 0A0E$_H$	32	1002 0094$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay MHD Setting Register	FLXA0FRMHDC	0000 0000$_H$	32	1002 0098$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay GTU Setting Register 1	FLXA0FRGTUC1	0000 0280$_H$	32	1002 00A0$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay GTU Setting Register 2	FLXA0FRGTUC2	0002 000A$_H$	32	1002 00A4$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay GTU Setting Register 3	FLXA0FRGTUC3	0202 0000$_H$	32	1002 00A8$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay GTU Setting Register 4	FLXA0FRGTUC4	0008 0007$_H$	32	1002 00AC$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay GTU Setting Register 5	FLXA0FRGTUC5	0E00 0000$_H$	32	1002 00B0$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay GTU Setting Register 6	FLXA0FRGTUC6	0002 0000$_H$	32	1002 00B4$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay GTU Setting Register 7	FLXA0FRGTUC7	0002 0004$_H$	32	1002 00B8$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay GTU Setting Register 8	FLXA0FRGTUC8	0000 0002$_H$	32	1002 00BC$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay GTU Setting Register 9	FLXA0FRGTUC9	0000 0101$_H$	32	1002 00C0$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay GTU Setting Register 10	FLXA0FRGTUC10	0002 0005$_H$	32	1002 00C4$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay GTU Setting Register 11	FLXA0FRGTUC11	0000 0000$_H$	32	1002 00C8$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay CC Status Vector Register	FLXA0FRCCSV	0010 4000$_H$	32	1002 0100$_H$	FlexRay	8, 16, 32

(2/201)

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
FLXA0	FlexRay CC Error Vector Register	FLXA0FRCCEV	0000 0000$_H$	32	1002 0104$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Slot Counter Value Register	FLXA0FRSCV	0000 0000$_H$	32	1002 0110$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay MT Value / Cycle Counter Value Register	FLXA0FRMTCCV	0000 0000$_H$	32	1002 0114$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Rate Correcting Value Register	FLXA0FRRCV	0000 0000$_H$	32	1002 0118$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Offset Correcting Value Register	FLXA0FROCV	0000 0000$_H$	32	1002 011C$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Sync Frame Status Register	FLXA0FRSFS	0000 0000$_H$	32	1002 0120$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Symbol Window /NIT Status Register	FLXA0FRSWNIT	0000 0000$_H$	32	1002 0124$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Aggregated Channel Status Register	FLXA0FRACS	0000 0000$_H$	32	1002 0128$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Even Sync ID Register 1	FLXA0FRESID1	0000 0000$_H$	32	1002 0130$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Even Sync ID Register 2	FLXA0FRESID2	0000 0000$_H$	32	1002 0134$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Even Sync ID Register 3	FLXA0FRESID3	0000 0000$_H$	32	1002 0138$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Even Sync ID Register 4	FLXA0FRESID4	0000 0000$_H$	32	1002 013C$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Even Sync ID Register 5	FLXA0FRESID5	0000 0000$_H$	32	1002 0140$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Even Sync ID Register 6	FLXA0FRESID6	0000 0000$_H$	32	1002 0144$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Even Sync ID Register 7	FLXA0FRESID7	0000 0000$_H$	32	1002 0148$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Even Sync ID Register 8	FLXA0FRESID8	0000 0000$_H$	32	1002 014C$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Even Sync ID Register 9	FLXA0FRESID9	0000 0000$_H$	32	1002 0150$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Even Sync ID Register 10	FLXA0FRESID10	0000 0000$_H$	32	1002 0154$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Even Sync ID Register 11	FLXA0FRESID11	0000 0000$_H$	32	1002 0158$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Even Sync ID Register 12	FLXA0FRESID12	0000 0000$_H$	32	1002 015C$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Even Sync ID Register 13	FLXA0FRESID13	0000 0000$_H$	32	1002 0160$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Even Sync ID Register 14	FLXA0FRESID14	0000 0000$_H$	32	1002 0164$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Even Sync ID Register 15	FLXA0FRESID15	0000 0000$_H$	32	1002 0168$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Odd Sync ID Register 1	FLXA0FROSID1	0000 0000$_H$	32	1002 0170$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Odd Sync ID Register 2	FLXA0FROSID2	0000 0000$_H$	32	1002 0174$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Odd Sync ID Register 3	FLXA0FROSID3	0000 0000$_H$	32	1002 0178$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Odd Sync ID Register 4	FLXA0FROSID4	0000 0000$_H$	32	1002 017C$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Odd Sync ID Register 5	FLXA0FROSID5	0000 0000$_H$	32	1002 0180$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Odd Sync ID Register 6	FLXA0FROSID6	0000 0000$_H$	32	1002 0184$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Odd Sync ID Register 7	FLXA0FROSID7	0000 0000$_H$	32	1002 0188$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Odd Sync ID Register 8	FLXA0FROSID8	0000 0000$_H$	32	1002 018C$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Odd Sync ID Register 9	FLXA0FROSID9	0000 0000$_H$	32	1002 0190$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Odd Sync ID Register 10	FLXA0FROSID10	0000 0000$_H$	32	1002 0194$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Odd Sync ID Register 11	FLXA0FROSID11	0000 0000$_H$	32	1002 0198$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Odd Sync ID Register 12	FLXA0FROSID12	0000 0000$_H$	32	1002 019C$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Odd Sync ID Register 13	FLXA0FROSID13	0000 0000$_H$	32	1002 01A0$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Odd Sync ID Register 14	FLXA0FROSID14	0000 0000$_H$	32	1002 01A4$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Odd Sync ID Register 15	FLXA0FROSID15	0000 0000$_H$	32	1002 01A8$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Network Management Vector Register 1	FLXA0FRNMV1	0000 0000$_H$	32	1002 01B0$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Network Management Vector Register 2	FLXA0FRNMV2	0000 0000$_H$	32	1002 01B4$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Network Management Vector Register 3	FLXA0FRNMV3	0000 0000$_H$	32	1002 01B8$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Message RAM Setting Register	FLXA0FRMRC	0180 0000$_H$	32	1002 0300$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay FIFO Rejection Filter Register	FLXA0FRFRF	0180 0000$_H$	32	1002 0304$_H$	FlexRay	8, 16, 32

(3/201)

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
FLXA0	FlexRay FIFO Rejection Filter Mask Register	FLXA0FRFRFM	0000 0000$_H$	32	1002 0308$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay FIFO Critical Level Register	FLXA0FRFCL	0000 0080$_H$	32	1002 030C$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Message Handler Status Register	FLXA0FRMHDS	0000 0080$_H$	32	1002 0310$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Last Dynamic Transmission Slot Register	FLXA0FRLDTS	0000 0000$_H$	32	1002 0314$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay FIFO Status Register	FLXA0FRFSR	0000 0000$_H$	32	1002 0318$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Message Handler Constraints Flag Register	FLXA0FRMHDF	0000 0000$_H$	32	1002 031C$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Transmission Request Register 1	FLXA0FRTXRQ1	0000 0000$_H$	32	1002 0320$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Transmission Request Register 2	FLXA0FRTXRQ2	0000 0000$_H$	32	1002 0324$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Transmission Request Register 3	FLXA0FRTXRQ3	0000 0000$_H$	32	1002 0328$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Transmission Request Register 4	FLXA0FRTXRQ4	0000 0000$_H$	32	1002 032C$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay new Data Register 1	FLXA0FRNDAT1	0000 0000$_H$	32	1002 0330$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay new Data Register 2	FLXA0FRNDAT2	0000 0000$_H$	32	1002 0334$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay new Data Register 3	FLXA0FRNDAT3	0000 0000$_H$	32	1002 0338$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay new Data Register 4	FLXA0FRNDAT4	0000 0000$_H$	32	1002 033C$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Message Buffer Status Changed Register 1	FLXA0FRMBSC1	0000 0000$_H$	32	1002 0340$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Message Buffer Status Changed Register 2	FLXA0FRMBSC2	0000 0000$_H$	32	1002 0344$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Message Buffer Status Changed Register 3	FLXA0FRMBSC3	0000 0000$_H$	32	1002 0348$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Message Buffer Status Changed Register 4	FLXA0FRMBSC4	0000 0000$_H$	32	1002 034C$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 1	FLXA0FRWRDS1	0000 0000$_H$	32	1002 0400$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 2	FLXA0FRWRDS2	0000 0000$_H$	32	1002 0404$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 3	FLXA0FRWRDS3	0000 0000$_H$	32	1002 0408$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 4	FLXA0FRWRDS4	0000 0000$_H$	32	1002 040C$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 5	FLXA0FRWRDS5	0000 0000$_H$	32	1002 0410$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 6	FLXA0FRWRDS6	0000 0000$_H$	32	1002 0414$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 7	FLXA0FRWRDS7	0000 0000$_H$	32	1002 0418$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 8	FLXA0FRWRDS8	0000 0000$_H$	32	1002 041C$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 9	FLXA0FRWRDS9	0000 0000$_H$	32	1002 0420$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 10	FLXA0FRWRDS10	0000 0000$_H$	32	1002 0424$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 11	FLXA0FRWRDS11	0000 0000$_H$	32	1002 0428$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 12	FLXA0FRWRDS12	0000 0000$_H$	32	1002 042C$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 13	FLXA0FRWRDS13	0000 0000$_H$	32	1002 0430$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 14	FLXA0FRWRDS14	0000 0000$_H$	32	1002 0434$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 15	FLXA0FRWRDS15	0000 0000$_H$	32	1002 0438$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 16	FLXA0FRWRDS16	0000 0000$_H$	32	1002 043C$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 17	FLXA0FRWRDS17	0000 0000$_H$	32	1002 0440$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 18	FLXA0FRWRDS18	0000 0000$_H$	32	1002 0444$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 19	FLXA0FRWRDS19	0000 0000$_H$	32	1002 0448$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 20	FLXA0FRWRDS20	0000 0000$_H$	32	1002 044C$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 21	FLXA0FRWRDS21	0000 0000$_H$	32	1002 0450$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 22	FLXA0FRWRDS22	0000 0000$_H$	32	1002 0454$_H$	FlexRay	8, 16, 32

(4/201)

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
FLXA0	FlexRay Data Selection Write Register 23	FLXA0FRWRDS23	0000 0000$_H$	32	1002 0458$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 24	FLXA0FRWRDS24	0000 0000$_H$	32	1002 045C$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 25	FLXA0FRWRDS25	0000 0000$_H$	32	1002 0460$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 26	FLXA0FRWRDS26	0000 0000$_H$	32	1002 0464$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 27	FLXA0FRWRDS27	0000 0000$_H$	32	1002 0468$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 28	FLXA0FRWRDS28	0000 0000$_H$	32	1002 046C$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 29	FLXA0FRWRDS29	0000 0000$_H$	32	1002 0470$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 30	FLXA0FRWRDS30	0000 0000$_H$	32	1002 0474$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 31	FLXA0FRWRDS31	0000 0000$_H$	32	1002 0478$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 32	FLXA0FRWRDS32	0000 0000$_H$	32	1002 047C$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 33	FLXA0FRWRDS33	0000 0000$_H$	32	1002 0480$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 34	FLXA0FRWRDS34	0000 0000$_H$	32	1002 0484$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 35	FLXA0FRWRDS35	0000 0000$_H$	32	1002 0488$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 36	FLXA0FRWRDS36	0000 0000$_H$	32	1002 048C$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 37	FLXA0FRWRDS37	0000 0000$_H$	32	1002 0490$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 38	FLXA0FRWRDS38	0000 0000$_H$	32	1002 0494$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 39	FLXA0FRWRDS39	0000 0000$_H$	32	1002 0498$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 40	FLXA0FRWRDS40	0000 0000$_H$	32	1002 049C$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 41	FLXA0FRWRDS41	0000 0000$_H$	32	1002 04A0$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 42	FLXA0FRWRDS42	0000 0000$_H$	32	1002 04A4$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 43	FLXA0FRWRDS43	0000 0000$_H$	32	1002 04A8$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 44	FLXA0FRWRDS44	0000 0000$_H$	32	1002 04AC$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 45	FLXA0FRWRDS45	0000 0000$_H$	32	1002 04B0$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 46	FLXA0FRWRDS46	0000 0000$_H$	32	1002 04B4$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 47	FLXA0FRWRDS47	0000 0000$_H$	32	1002 04B8$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 48	FLXA0FRWRDS48	0000 0000$_H$	32	1002 04BC$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 49	FLXA0FRWRDS49	0000 0000$_H$	32	1002 04C0$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 50	FLXA0FRWRDS50	0000 0000$_H$	32	1002 04C4$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 51	FLXA0FRWRDS51	0000 0000$_H$	32	1002 04C8$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 52	FLXA0FRWRDS52	0000 0000$_H$	32	1002 04CC$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 53	FLXA0FRWRDS53	0000 0000$_H$	32	1002 04D0$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 54	FLXA0FRWRDS54	0000 0000$_H$	32	1002 04D4$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 55	FLXA0FRWRDS55	0000 0000$_H$	32	1002 04D8$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 56	FLXA0FRWRDS56	0000 0000$_H$	32	1002 04DC$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 57	FLXA0FRWRDS57	0000 0000$_H$	32	1002 04E0$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 58	FLXA0FRWRDS58	0000 0000$_H$	32	1002 04E4$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 59	FLXA0FRWRDS59	0000 0000$_H$	32	1002 04E8$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 60	FLXA0FRWRDS60	0000 0000$_H$	32	1002 04EC$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 61	FLXA0FRWRDS61	0000 0000$_H$	32	1002 04F0$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 62	FLXA0FRWRDS62	0000 0000$_H$	32	1002 04F4$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 63	FLXA0FRWRDS63	0000 0000$_H$	32	1002 04F8$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Write Register 64	FLXA0FRWRDS64	0000 0000$_H$	32	1002 04FC$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Header Selection Write Register 1	FLXA0FRWRHS1	0000 0000$_H$	32	1002 0500$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Header Selection Write Register 2	FLXA0FRWRHS2	0000 0000$_H$	32	1002 0504$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Header Selection Write Register 3	FLXA0FRWRHS3	0000 0000$_H$	32	1002 0508$_H$	FlexRay	8, 16, 32

(5/201)

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
FLXA0	FlexRay Input Buffer Command Mask Register	FLXA0FRIBCM	0000 0000$_H$	32	1002 0510$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Input Buffer Command Request Register	FLXA0FRIBCR	0000 0000$_H$	32	1002 0514$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 1	FLXA0FRRDDS1	0000 0000$_H$	32	1002 0600$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 2	FLXA0FRRDDS2	0000 0000$_H$	32	1002 0604$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 3	FLXA0FRRDDS3	0000 0000$_H$	32	1002 0608$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 4	FLXA0FRRDDS4	0000 0000$_H$	32	1002 060C$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 5	FLXA0FRRDDS5	0000 0000$_H$	32	1002 0610$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 6	FLXA0FRRDDS6	0000 0000$_H$	32	1002 0614$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 7	FLXA0FRRDDS7	0000 0000$_H$	32	1002 0618$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 8	FLXA0FRRDDS8	0000 0000$_H$	32	1002 061C$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 9	FLXA0FRRDDS9	0000 0000$_H$	32	1002 0620$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 10	FLXA0FRRDDS10	0000 0000$_H$	32	1002 0624$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 11	FLXA0FRRDDS11	0000 0000$_H$	32	1002 0628$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 12	FLXA0FRRDDS12	0000 0000$_H$	32	1002 062C$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 13	FLXA0FRRDDS13	0000 0000$_H$	32	1002 0630$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 14	FLXA0FRRDDS14	0000 0000$_H$	32	1002 0634$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 15	FLXA0FRRDDS15	0000 0000$_H$	32	1002 0638$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 16	FLXA0FRRDDS16	0000 0000$_H$	32	1002 063C$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 17	FLXA0FRRDDS17	0000 0000$_H$	32	1002 0640$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 18	FLXA0FRRDDS18	0000 0000$_H$	32	1002 0644$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 19	FLXA0FRRDDS19	0000 0000$_H$	32	1002 0648$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 20	FLXA0FRRDDS20	0000 0000$_H$	32	1002 064C$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 21	FLXA0FRRDDS21	0000 0000$_H$	32	1002 0650$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 22	FLXA0FRRDDS22	0000 0000$_H$	32	1002 0654$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 23	FLXA0FRRDDS23	0000 0000$_H$	32	1002 0658$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 24	FLXA0FRRDDS24	0000 0000$_H$	32	1002 065C$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 25	FLXA0FRRDDS25	0000 0000$_H$	32	1002 0660$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 26	FLXA0FRRDDS26	0000 0000$_H$	32	1002 0664$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 27	FLXA0FRRDDS27	0000 0000$_H$	32	1002 0668$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 28	FLXA0FRRDDS28	0000 0000$_H$	32	1002 066C$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 29	FLXA0FRRDDS29	0000 0000$_H$	32	1002 0670$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 30	FLXA0FRRDDS30	0000 0000$_H$	32	1002 0674$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 31	FLXA0FRRDDS31	0000 0000$_H$	32	1002 0678$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 32	FLXA0FRRDDS32	0000 0000$_H$	32	1002 067C$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 33	FLXA0FRRDDS33	0000 0000$_H$	32	1002 0680$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 34	FLXA0FRRDDS34	0000 0000$_H$	32	1002 0684$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 35	FLXA0FRRDDS35	0000 0000$_H$	32	1002 0688$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 36	FLXA0FRRDDS36	0000 0000$_H$	32	1002 068C$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 37	FLXA0FRRDDS37	0000 0000$_H$	32	1002 0690$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 38	FLXA0FRRDDS38	0000 0000$_H$	32	1002 0694$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 39	FLXA0FRRDDS39	0000 0000$_H$	32	1002 0698$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 40	FLXA0FRRDDS40	0000 0000$_H$	32	1002 069C$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 41	FLXA0FRRDDS41	0000 0000$_H$	32	1002 06A0$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 42	FLXA0FRRDDS42	0000 0000$_H$	32	1002 06A4$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 43	FLXA0FRRDDS43	0000 0000$_H$	32	1002 06A8$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 44	FLXA0FRRDDS44	0000 0000$_H$	32	1002 06AC$_H$	FlexRay	8, 16, 32

(6/201)

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
FLXA0	FlexRay Data Selection Read Register 45	FLXA0FRRDDS45	0000 0000$_H$	32	1002 06B0$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 46	FLXA0FRRDDS46	0000 0000$_H$	32	1002 06B4$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 47	FLXA0FRRDDS47	0000 0000$_H$	32	1002 06B8$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 48	FLXA0FRRDDS48	0000 0000$_H$	32	1002 06BC$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 49	FLXA0FRRDDS49	0000 0000$_H$	32	1002 06C0$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 50	FLXA0FRRDDS50	0000 0000$_H$	32	1002 06C4$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 51	FLXA0FRRDDS51	0000 0000$_H$	32	1002 06C8$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 52	FLXA0FRRDDS52	0000 0000$_H$	32	1002 06CC$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 53	FLXA0FRRDDS53	0000 0000$_H$	32	1002 06D0$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 54	FLXA0FRRDDS54	0000 0000$_H$	32	1002 06D4$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 55	FLXA0FRRDDS55	0000 0000$_H$	32	1002 06D8$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 56	FLXA0FRRDDS56	0000 0000$_H$	32	1002 06DC$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 57	FLXA0FRRDDS57	0000 0000$_H$	32	1002 06E0$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 58	FLXA0FRRDDS58	0000 0000$_H$	32	1002 06E4$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 59	FLXA0FRRDDS59	0000 0000$_H$	32	1002 06E8$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 60	FLXA0FRRDDS60	0000 0000$_H$	32	1002 06EC$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 61	FLXA0FRRDDS61	0000 0000$_H$	32	1002 06F0$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 62	FLXA0FRRDDS62	0000 0000$_H$	32	1002 06F4$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 63	FLXA0FRRDDS63	0000 0000$_H$	32	1002 06F8$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Data Selection Read Register 64	FLXA0FRRDDS64	0000 0000$_H$	32	1002 06FC$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Header selection read Register 1	FLXA0FRRDHS1	0000 0000$_H$	32	1002 0700$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Header selection read Register 2	FLXA0FRRDHS2	0000 0000$_H$	32	1002 0704$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Header selection read Register 3	FLXA0FRRDHS3	0000 0000$_H$	32	1002 0708$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Message Buffer Status Register	FLXA0FRMBS	0000 0000$_H$	32	1002 070C$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Output Buffer Command Mask Register	FLXA0FROBCM	0000 0000$_H$	32	1002 0710$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Output Buffer Command Request Register	FLXA0FROBCR	0000 0000$_H$	32	1002 0714$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Input Transfer Setting Register	FLXA0FRITC	0000 0000$_H$	32	1002 0800$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Output Transfer Setting Register	FLXA0FROTC	0000 0000$_H$	32	1002 0804$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Input Pointer Table Base Address Register	FLXA0FRIBA	0000 0000$_H$	32	1002 0808$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay FIFO Pointer Table Base Address Register	FLXA0FRFBA	0000 0000$_H$	32	1002 080C$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Output Pointer Table Base Address Register	FLXA0FROBA	0000 0000$_H$	32	1002 0810$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Input Queue Control Register	FLXA0FRIQC	0000 0000$_H$	32	1002 0814$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay User Input Transfer Request Register	FLXA0FRUIR	0000 0000$_H$	32	1002 0818$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay User Output Transfer Request Register	FLXA0FRUOR	0000 0000$_H$	32	1002 081C$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Input Transfer Status Register	FLXA0FRITS	0000 0000$_H$	32	1002 0820$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Output Transfer Status Register	FLXA0FROTS	0000 0000$_H$	32	1002 0824$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Access Error Status Register	FLXA0FRAES	0000 0000$_H$	32	1002 0828$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Access Error Address Register	FLXA0FRAEA	0000 0000$_H$	32	1002 082C$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Message Data Available Register 0	FLXA0FRDA0	0000 0000$_H$	32	1002 0830$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Message Data Available Register 1	FLXA0FRDA1	0000 0000$_H$	32	1002 0834$_H$	FlexRay	8, 16, 32

(7/201)

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
FLXA0	FlexRay Message Data Available Register 2	FLXA0FRDA2	0000 0000$_H$	32	1002 0838$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Message Data Available Register 3	FLXA0FRDA3	0000 0000$_H$	32	1002 083C$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay H-Bus Setting Register	FLXA0FRAHBC	0000 0000$_H$	32	1002 0840$_H$	FlexRay	8, 16, 32
FLXA0	FlexRay Timer 2 Setting Register	FLXA0FRT2C	0000 0000$_H$	32	1002 0844$_H$	FlexRay	8, 16, 32
AUD	AUDR Configuration Information Retention Register	AUDISR	000X$_H$	16	FA00 5000$_H$	CPU(DEBUG)	16
AUD	AUDR Message Board Register	AUDMBR	0000$_H$	16	FA00 5004$_H$	CPU(DEBUG)	16
AUD	AUDR Message Board Register	AUDMBRC	0000$_H$	16	FA00 5008$_H$	CPU(DEBUG)	16
PORT	Port Register	P0	0000$_H$	16	FF61 0000$_H$	3	16
PORT	Port Set / Reset Register	PSR0	0000 0000$_H$	32	FF61 0004$_H$	3	32
PORT	Port NOT Register	PNOT0	0000$_H$	16	FF61 0008$_H$	3	16
PORT	Port Pin Read Register	PPR0	0000$_H$	16	FF61 000C$_H$	3	16
PORT	Port Mode Register	PM0	FFFF$_H$	16	FF61 0010$_H$	3	16
PORT	Port Mode Control Register	PMC0	0000$_H$	16	FF61 0014$_H$	3	16
PORT	Port Function Control Register	PFC0	0000$_H$	16	FF61 0018$_H$	3	16
PORT	Port Function Control Expansion Register	PFCE0	0000$_H$	16	FF61 001C$_H$	3	16
PORT	Port Mode Set / Reset Register	PMSR0	0000 FFFF$_H$	32	FF61 0020$_H$	3	32
PORT	Port Mode Control Set / Reset Register	PMCSR0	0000 0000$_H$	32	FF61 0024$_H$	3	32
PORT	Port Function Control Additional Expansion Register	PFCAE0	0000$_H$	16	FF61 0028$_H$	3	16
PORT	Port Output Level Inversion Register	PINV0	0000 0000$_H$	32	FF61 0030$_H$	3	32
PORT	Port Register	P1	0000$_H$	16	FF61 0040$_H$	3	16
PORT	Port Set/Reset Register	PSR1	0000 0000$_H$	32	FF61 0044$_H$	3	32
PORT	Port NOT Register	PNOT1	0000$_H$	16	FF61 0048$_H$	3	16
PORT	Port Pin Read Register	PPR1	0000$_H$	16	FF61 004C$_H$	3	16
PORT	Port Mode Register	PM1	FFFF$_H$	16	FF61 0050$_H$	3	16
PORT	Port Mode Control Register	PMC1	0000$_H$	16	FF61 0054$_H$	3	16
PORT	Port Function Control Register	PFC1	0000$_H$	16	FF61 0058$_H$	3	16
PORT	Port Function Control Expansion Register	PFCE1	0000$_H$	16	FF61 005C$_H$	3	16
PORT	Port Mode Set / Reset Register	PMSR1	0000 FFFF$_H$	32	FF61 0060$_H$	3	32
PORT	Port Mode Control Set / Reset Register	PMCSR1	0000 0000$_H$	32	FF61 0064$_H$	3	32
PORT	Port Function Control Additional Expansion Register	PFCAE1	0000$_H$	16	FF61 0068$_H$	3	16
PORT	Port Output Level Inversion Register	PINV1	0000 0000$_H$	32	FF61 0070$_H$	3	32
PORT	Port Register	P2	0000$_H$	16	FF61 0080$_H$	3	16
PORT	Port Set / Reset Register	PSR2	0000 0000$_H$	32	FF61 0084$_H$	3	32
PORT	Port NOT Register	PNOT2	0000$_H$	16	FF61 0088$_H$	3	16
PORT	Port Pin Read Register	PPR2	0000$_H$	16	FF61 008C$_H$	3	16
PORT	Port Mode Register	PM2	FFFF$_H$	16	FF61 0090$_H$	3	16
PORT	Port Mode Control Register	PMC2	0000$_H$	16	FF61 0094$_H$	3	16
PORT	Port Function Control Register	PFC2	0000$_H$	16	FF61 0098$_H$	3	16
PORT	Port Function Control Expansion Register	PFCE2	0000$_H$	16	FF61 009C$_H$	3	16
PORT	Port Mode Set / Reset Register	PMSR2	0000 FFFF$_H$	32	FF61 00A0$_H$	3	32
PORT	Port Mode Control Set / Reset Register	PMCSR2	0000 0000$_H$	32	FF61 00A4$_H$	3	32

(8/201)

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
PORT	Port Function Control Additional Expansion Register	PFCAE2	0000_H	16	FF61 00A8$_H$	3	16
PORT	Port Output Level Inversion Register	PINV2	0000 0000_H	32	FF61 00B0$_H$	3	32
PORT	Port Register	P3	0000_H	16	FF61 00C0$_H$	3	16
PORT	Port Set / Reset Register	PSR3	0000 0000_H	32	FF61 00C4$_H$	3	32
PORT	Port NOT Register	PNOT3	0000_H	16	FF61 00C8$_H$	3	16
PORT	Port Pin Read Register	PPR3	0000_H	16	FF61 00CC$_H$	3	16
PORT	Port Mode Register	PM3	$FFFF_H$	16	FF61 00D0$_H$	3	16
PORT	Port Mode Control Register	PMC3	0000_H	16	FF61 00D4$_H$	3	16
PORT	Port Function Control Register	PFC3	0000_H	16	FF61 00D8$_H$	3	16
PORT	Port Function Control Expansion Register	PFCE3	0000_H	16	FF61 00DC$_H$	3	16
PORT	Port Mode Set / Reset Register	PMSR3	0000 $FFFF_H$	32	FF61 00E0$_H$	3	32
PORT	Port Mode Control Set / Reset Register	PMCSR3	0000 0000_H	32	FF61 00E4$_H$	3	32
PORT	Port Function Control Additional Expansion Register	PFCAE3	0000_H	16	FF61 00E8$_H$	3	16
PORT	Port Output Level Inversion Register	PINV3	0000 0000_H	32	FF61 00F0$_H$	3	32
PORT	Port Register	P4	0000_H	16	FF61 0100$_H$	3	16
PORT	Port Set / Reset Register	PSR4	0000 0000_H	32	FF61 0104$_H$	3	32
PORT	Port NOT Register	PNOT4	0000_H	16	FF61 0108$_H$	3	16
PORT	Port Pin Read Register	PPR4	0000_H	16	FF61 010C$_H$	3	16
PORT	Port Mode Register	PM4	$FFFF_H$	16	FF61 0110$_H$	3	16
PORT	Port Mode Control Register	PMC4	0000_H	16	FF61 0114$_H$	3	16
PORT	Port Function Control Register	PFC4	0000_H	16	FF61 0118$_H$	3	16
PORT	Port Function Control Expansion Register	PFCE4	0000_H	16	FF61 011C$_H$	3	16
PORT	Port Mode Set / Reset Register	PMSR4	0000 $FFFF_H$	32	FF61 0120$_H$	3	32
PORT	Port Mode Control Set / Reset Register	PMCSR4	0000 0000_H	32	FF61 0124$_H$	3	32
PORT	Port Function Control Additional Expansion Register	PFCAE4	0000_H	16	FF61 0128$_H$	3	16
PORT	Port Output Level Inversion Register	PINV4	0000 0000_H	32	FF61 0130$_H$	3	32
PORT	Port Register	P5	0000_H	16	FF61 0140$_H$	3	16
PORT	Port Set / Reset Register	PSR5	0000 0000_H	32	FF61 0144$_H$	3	32
PORT	Port NOT Register	PNOT5	0000_H	16	FF61 0148$_H$	3	16
PORT	Port Pin Read Register	PPR5	0000_H	16	FF61 014C$_H$	3	16
PORT	Port Mode Register	PM5	$FFFF_H$	16	FF61 0150$_H$	3	16
PORT	Port Mode Control Register	PMC5	0000_H	16	FF61 0154$_H$	3	16
PORT	Port Function Control Register	PFC5	0000_H	16	FF61 0158$_H$	3	16
PORT	Port Function Control Expansion Register	PFCE5	0000_H	16	FF61 015C$_H$	3	16
PORT	Port Mode Set / Reset Register	PMSR5	0000 $FFFF_H$	32	FF61 0160$_H$	3	32
PORT	Port Mode Control Set / Reset Register	PMCSR5	0000 0000_H	32	FF61 0164$_H$	3	32
PORT	Port Function Control Additional Expansion Register	PFCAE5	0000_H	16	FF61 0168$_H$	3	16
PORT	Port Output Level Inversion Register	PINV5	0000 0000_H	32	FF61 0170$_H$	3	32
PORT	Port Register	P6	0000_H	16	FF61 0180$_H$	3	16
PORT	Port Set / Reset Register	PSR6	0000 0000_H	32	FF61 0184$_H$	3	32
PORT	Port NOT Register	PNOT6	0000_H	16	FF61 0188$_H$	3	16
PORT	Port Pin Read Register	PPR6	0000_H	16	FF61 018C$_H$	3	16
PORT	Port Mode Register	PM6	$FFFF_H$	16	FF61 0190$_H$	3	16

(9/201)

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
PORT	Port Mode Control Register	PMC6	0000_H	16	FF61 0194_H	3	16
PORT	Port Function Control Register	PFC6	0000_H	16	FF61 0198_H	3	16
PORT	Port Function Control Expansion Register	PFCE6	0000_H	16	FF61 $019C_H$	3	16
PORT	Port Mode Set / Reset Register	PMSR6	0000 $FFFF_H$	32	FF61 $01A0_H$	3	32
PORT	Port Mode Control Set / Reset Register	PMCSR6	0000 0000_H	32	FF61 $01A4_H$	3	32
PORT	Port Function Control Additional Expansion Register	PFCAE6	0000_H	16	FF61 $01A8_H$	3	16
PORT	Port Output Level Inversion Register	PINV6	0000 0000_H	32	FF61 $01B0_H$	3	32
PORT	Port Register	P7	0000_H	16	FF61 $01C0_H$	3	16
PORT	Port Set / Reset Register	PSR7	0000 0000_H	32	FF61 $01C4_H$	3	32
PORT	Port NOT Register	PNOT7	0000_H	16	FF61 $01C8_H$	3	16
PORT	Port Pin Read Register	PPR7	0000_H	16	FF61 $01CC_H$	3	16
PORT	Port Mode Register	PM7	$FFFF_H$	16	FF61 $01D0_H$	3	16
PORT	Port Mode Control Register	PMC7	0000_H	16	FF61 $01D4_H$	3	16
PORT	Port Function Control Register	PFC7	0000_H	16	FF61 $01D8_H$	3	16
PORT	Port Function Control Expansion Register	PFCE7	0000_H	16	FF61 $01DC_H$	3	16
PORT	Port Mode Set / Reset Register	PMSR7	0000 $FFFF_H$	32	FF61 $01E0_H$	3	32
PORT	Port Mode Control Set / Reset Register	PMCSR7	0000 0000_H	32	FF61 $01E4_H$	3	32
PORT	Port Function Control Additional Expansion Register	PFCAE7	0000_H	16	FF61 $01E8_H$	3	16
PORT	Port Output Level Inversion Register	PINV7	0000 0000_H	32	FF61 $01F0_H$	3	32
PORT	Port Register	P8	0000_H	16	FF61 0200_H	3	16
PORT	Port Set / Reset Register	PSR8	0000 0000_H	32	FF61 0204_H	3	32
PORT	Port NOT Register	PNOT8	0000_H	16	FF61 0208_H	3	16
PORT	Port Pin Read Register	PPR8	0000_H	16	FF61 $020C_H$	3	16
PORT	Port Mode Register	PM8	$FFFF_H$	16	FF61 0210_H	3	16
PORT	Port Mode Control Register	PMC8	0000_H	16	FF61 0214_H	3	16
PORT	Port Function Control Register	PFC8	0000_H	16	FF61 0218_H	3	16
PORT	Port Function Control Expansion Register	PFCE8	0000_H	16	FF61 $021C_H$	3	16
PORT	Port Mode Set / Reset Register	PMSR8	0000 $FFFF_H$	32	FF61 0220_H	3	32
PORT	Port Mode Control Set / Reset Register	PMCSR8	0000 0000_H	32	FF61 0224_H	3	32
PORT	Port Function Control Additional Expansion Register	PFCAE8	0000_H	16	FF61 0228_H	3	16
PORT	Port Output Level Inversion Register	PINV8	0000 0000_H	32	FF61 0230_H	3	32
PORT	Port Register	P10	0000_H	16	FF61 0280_H	3	16
PORT	Port Set / Reset Register	PSR10	0000 0000_H	32	FF61 0284_H	3	32
PORT	Port NOT Register	PNOT10	0000_H	16	FF61 0288_H	3	16
PORT	Port Pin Read Register	PPR10	0000_H	16	FF61 $028C_H$	3	16
PORT	Port Mode Register	PM10	$FFFF_H$	16	FF61 0290_H	3	16
PORT	Port Mode Control Register	PMC10	0000_H	16	FF61 0294_H	3	16
PORT	Port Function Control Register	PFC10	0000_H	16	FF61 0298_H	3	16
PORT	Port Function Control Expansion Register	PFCE10	0000_H	16	FF61 $029C_H$	3	16
PORT	Port Mode Set / Reset Register	PMSR10	0000 $FFFF_H$	32	FF61 $02A0_H$	3	32
PORT	Port Mode Control Set / Reset Register	PMCSR10	0000 0000_H	32	FF61 $02A4_H$	3	32
PORT	Port Function Control Additional Expansion Register	PFCAE10	0000_H	16	FF61 $02A8_H$	3	16

(10/201)

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
PORT	Port Output Level Inversion Register	PINV10	0000 0000$_H$	32	FF61 02B0$_H$	3	32
PORT	Port Register	P11	0000$_H$	16	FF61 02C0$_H$	3	16
PORT	Port Set / Reset Register	PSR11	0000 0000$_H$	32	FF61 02C4$_H$	3	32
PORT	Port NOT Register	PNOT11	0000$_H$	16	FF61 02C8$_H$	3	16
PORT	Port Pin Read Register	PPR11	0000$_H$	16	FF61 02CC$_H$	3	16
PORT	Port Mode Register	PM11	FFFF$_H$	16	FF61 02D0$_H$	3	16
PORT	Port Mode Control Register	PMC11	0000$_H$	16	FF61 02D4$_H$	3	16
PORT	Port Function Control Register	PFC11	0000$_H$	16	FF61 02D8$_H$	3	16
PORT	Port Function Control Expansion Register	PFCE11	0000$_H$	16	FF61 02DC$_H$	3	16
PORT	Port Mode Set / Reset Register	PMSR11	0000 FFFF$_H$	32	FF61 02E0$_H$	3	32
PORT	Port Mode Control Set / Reset Register	PMCSR11	0000 0000$_H$	32	FF61 02E4$_H$	3	32
PORT	Port Output Level Inversion Register	PINV11	0000 0000$_H$	32	FF61 02F0$_H$	3	32
PORT	Port Register	P13	0000$_H$	16	FF61 0340$_H$	3	16
PORT	Port Set / Reset Register	PSR13	0000 0000$_H$	32	FF61 0344$_H$	3	32
PORT	Port NOT Register	PNOT13	0000$_H$	16	FF61 0348$_H$	3	16
PORT	Port Pin Read Register	PPR13	0000$_H$	16	FF61 034C$_H$	3	16
PORT	Port Mode Register	PM13	FFFF$_H$	16	FF61 0350$_H$	3	16
PORT	Port Mode Control Register	PMC13	0000$_H$	16	FF61 0354$_H$	3	16
PORT	Port Function Control Register	PFC13	0000$_H$	16	FF61 0358$_H$	3	16
PORT	Port Function Control Expansion Register	PFCE13	0000$_H$	16	FF61 035C$_H$	3	16
PORT	Port Mode Set / Reset Register	PMSR13	0000 FFFF$_H$	32	FF61 0360$_H$	3	32
PORT	Port Mode Control Set / Reset Register	PMCSR13	0000 0000$_H$	32	FF61 0364$_H$	3	32
PORT	Port Function Control Additional Expansion Register	PFCAE13	0000$_H$	16	FF61 0368$_H$	3	16
PORT	Port Output Level Inversion Register	PINV13	0000 0000$_H$	32	FF61 0370$_H$	3	32
PORT	Port Pin Read Register	PPR14	0000$_H$	16	FF61 038C$_H$	3	16
PORT	Port Mode Control Register	PMC14	0000$_H$	16	FF61 0394$_H$	3	16
PORT	Port Function Control Register	PFC14	0000$_H$	16	FF61 0398$_H$	3	16
PORT	Port Mode Control Set / Reset Register	PMCSR14	0000 0000$_H$	32	FF61 03A4$_H$	3	32
PORT	Port Pin Read Register	PPR15	0000$_H$	16	FF61 03CC$_H$	3	16
PORT	Port Mode Control Register	PMC15	0000$_H$	16	FF61 03D4$_H$	3	16
PORT	Port Function Control Register	PFC15	0000$_H$	16	FF61 03D8$_H$	3	16
PORT	Port Mode Control Set / Reset Register	PMCSR15	0000 0000$_H$	32	FF61 03E4$_H$	3	32
PORT	Port Pin Read Register	PPR16	0000$_H$	16	FF61 040C$_H$	3	16
PORT	Port Mode Register	PM17	FFFF$_H$	16	FF61 0450$_H$	3	16
PORT	Port Mode Set / Reset Register	PMSR17	0000 FFFF$_H$	32	FF61 0460$_H$	3	32
PORT	Port Control Register	PCR0_0	0000 0010$_H$	32	FF61 2000$_H$	3	32
PORT	Port Control Register	PCR0_1	0000 0010$_H$	32	FF61 2004$_H$	3	32
PORT	Port Control Register	PCR0_2	0000 0010$_H$	32	FF61 2008$_H$	3	32
PORT	Port Control Register	PCR0_3	0000 0010$_H$	32	FF61 200C$_H$	3	32
PORT	Port Control Register	PCR0_4	0000 0010$_H$	32	FF61 2010$_H$	3	32
PORT	Port Control Register	PCR0_5	0000 0010$_H$	32	FF61 2014$_H$	3	32
PORT	Port Control Register	PCR0_6	0000 0010$_H$	32	FF61 2018$_H$	3	32
PORT	Port Control Register	PCR0_7	0000 0010$_H$	32	FF61 201C$_H$	3	32
PORT	Port Control Register	PCR0_8	0000 0010$_H$	32	FF61 2020$_H$	3	32
PORT	Port Control Register	PCR0_9	0000 0010$_H$	32	FF61 2024$_H$	3	32

(11/201)

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
PORT	Port Control Register	PCR0_10	0000 0010$_H$	32	FF61 2028$_H$	3	32
PORT	Port Control Register	PCR0_11	0000 0010$_H$	32	FF61 202C$_H$	3	32
PORT	Port Control Register	PCR0_12	0000 0010$_H$	32	FF61 2030$_H$	3	32
PORT	Port Control Register	PCR0_13	0000 0010$_H$	32	FF61 2034$_H$	3	32
PORT	Port Control Register	PCR0_14	0000 0010$_H$	32	FF61 2038$_H$	3	32
PORT	Port Control Register	PCR1_0	0000 0010$_H$	32	FF61 2040$_H$	3	32
PORT	Port Control Register	PCR1_1	0000 0010$_H$	32	FF61 2044$_H$	3	32
PORT	Port Control Register	PCR1_2	0000 0010$_H$	32	FF61 2048$_H$	3	32
PORT	Port Control Register	PCR1_3	0000 0010$_H$	32	FF61 204C$_H$	3	32
PORT	Port Control Register	PCR1_4	0000 0010$_H$	32	FF61 2050$_H$	3	32
PORT	Port Control Register	PCR1_5	0000 0010$_H$	32	FF61 2054$_H$	3	32
PORT	Port Control Register	PCR1_6	0000 0010$_H$	32	FF61 2058$_H$	3	32
PORT	Port Control Register	PCR1_7	0000 0010$_H$	32	FF61 205C$_H$	3	32
PORT	Port Control Register	PCR1_8	0000 0010$_H$	32	FF61 2060$_H$	3	32
PORT	Port Control Register	PCR1_9	0000 0010$_H$	32	FF61 2064$_H$	3	32
PORT	Port Control Register	PCR1_10	0000 0010$_H$	32	FF61 2068$_H$	3	32
PORT	Port Control Register	PCR1_11	0000 0010$_H$	32	FF61 206C$_H$	3	32
PORT	Port Control Register	PCR1_12	0000 0010$_H$	32	FF61 2070$_H$	3	32
PORT	Port Control Register	PCR1_13	0000 0010$_H$	32	FF61 2074$_H$	3	32
PORT	Port Control Register	PCR1_14	0000 0010$_H$	32	FF61 2078$_H$	3	32
PORT	Port Control Register	PCR1_15	0000 0010$_H$	32	FF61 207C$_H$	3	32
PORT	Port Control Register	PCR2_0	0000 0010$_H$	32	FF61 2080$_H$	3	32
PORT	Port Control Register	PCR2_1	0000 0010$_H$	32	FF61 2084$_H$	3	32
PORT	Port Control Register	PCR2_2	0000 0010$_H$	32	FF61 2088$_H$	3	32
PORT	Port Control Register	PCR2_3	0000 0010$_H$	32	FF61 208C$_H$	3	32
PORT	Port Control Register	PCR2_4	0000 0010$_H$	32	FF61 2090$_H$	3	32
PORT	Port Control Register	PCR2_5	0000 0010$_H$	32	FF61 2094$_H$	3	32
PORT	Port Control Register	PCR2_6	0000 0010$_H$	32	FF61 2098$_H$	3	32
PORT	Port Control Register	PCR2_7	0000 0010$_H$	32	FF61 209C$_H$	3	32
PORT	Port Control Register	PCR3_0	0000 0010$_H$	32	FF61 20C0$_H$	3	32
PORT	Port Control Register	PCR3_1	0000 0010$_H$	32	FF61 20C4$_H$	3	32
PORT	Port Control Register	PCR3_2	0000 0010$_H$	32	FF61 20C8$_H$	3	32
PORT	Port Control Register	PCR3_3	0000 0010$_H$	32	FF61 20CC$_H$	3	32
PORT	Port Control Register	PCR3_4	0000 0010$_H$	32	FF61 20D0$_H$	3	32
PORT	Port Control Register	PCR3_5	0000 0010$_H$	32	FF61 20D4$_H$	3	32
PORT	Port Control Register	PCR3_6	0000 0010$_H$	32	FF61 20D8$_H$	3	32
PORT	Port Control Register	PCR3_7	0000 0010$_H$	32	FF61 20DC$_H$	3	32
PORT	Port Control Register	PCR4_0	0000 0010$_H$	32	FF61 2100$_H$	3	32
PORT	Port Control Register	PCR4_1	0000 0010$_H$	32	FF61 2104$_H$	3	32
PORT	Port Control Register	PCR4_2	0000 0010$_H$	32	FF61 2108$_H$	3	32
PORT	Port Control Register	PCR4_3	0000 0010$_H$	32	FF61 210C$_H$	3	32
PORT	Port Control Register	PCR4_4	0000 0010$_H$	32	FF61 2110$_H$	3	32
PORT	Port Control Register	PCR4_5	0000 0010$_H$	32	FF61 2114$_H$	3	32
PORT	Port Control Register	PCR4_6	0000 0010$_H$	32	FF61 2118$_H$	3	32
PORT	Port Control Register	PCR4_7	0000 0010$_H$	32	FF61 211C$_H$	3	32
PORT	Port Control Register	PCR4_8	0000 0010$_H$	32	FF61 2120$_H$	3	32
PORT	Port Control Register	PCR4_9	0000 0010$_H$	32	FF61 2124$_H$	3	32

(12/201)

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
PORT	Port Control Register	PCR4_10	0000 0010$_H$	32	FF61 2128$_H$	3	32
PORT	Port Control Register	PCR4_11	0000 0010$_H$	32	FF61 212C$_H$	3	32
PORT	Port Control Register	PCR4_12	0000 0010$_H$	32	FF61 2130$_H$	3	32
PORT	Port Control Register	PCR4_13	0000 0010$_H$	32	FF61 2134$_H$	3	32
PORT	Port Control Register	PCR4_14	0000 0010$_H$	32	FF61 2138$_H$	3	32
PORT	Port Control Register	PCR4_15	0000 0010$_H$	32	FF61 213C$_H$	3	32
PORT	Port Control Register	PCR5_0	0000 0010$_H$	32	FF61 2140$_H$	3	32
PORT	Port Control Register	PCR5_1	0000 0010$_H$	32	FF61 2144$_H$	3	32
PORT	Port Control Register	PCR5_2	0000 0010$_H$	32	FF61 2148$_H$	3	32
PORT	Port Control Register	PCR5_3	0000 0010$_H$	32	FF61 214C$_H$	3	32
PORT	Port Control Register	PCR5_4	0000 0010$_H$	32	FF61 2150$_H$	3	32
PORT	Port Control Register	PCR5_5	0000 0010$_H$	32	FF61 2154$_H$	3	32
PORT	Port Control Register	PCR5_6	0000 0010$_H$	32	FF61 2158$_H$	3	32
PORT	Port Control Register	PCR5_7	0000 0010$_H$	32	FF61 215C$_H$	3	32
PORT	Port Control Register	PCR5_8	0000 0010$_H$	32	FF61 2160$_H$	3	32
PORT	Port Control Register	PCR5_9	0000 0010$_H$	32	FF61 2164$_H$	3	32
PORT	Port Control Register	PCR6_0	0000 0010$_H$	32	FF61 2180$_H$	3	32
PORT	Port Control Register	PCR6_1	0000 0010$_H$	32	FF61 2184$_H$	3	32
PORT	Port Control Register	PCR6_2	0000 0010$_H$	32	FF61 2188$_H$	3	32
PORT	Port Control Register	PCR6_3	0000 0010$_H$	32	FF61 218C$_H$	3	32
PORT	Port Control Register	PCR6_4	0000 0010$_H$	32	FF61 2190$_H$	3	32
PORT	Port Control Register	PCR6_5	0000 0010$_H$	32	FF61 2194$_H$	3	32
PORT	Port Control Register	PCR6_6	0000 0010$_H$	32	FF61 2198$_H$	3	32
PORT	Port Control Register	PCR6_7	0000 0010$_H$	32	FF61 219C$_H$	3	32
PORT	Port Control Register	PCR6_8	0000 0010$_H$	32	FF61 21A0$_H$	3	32
PORT	Port Control Register	PCR6_9	0000 0010$_H$	32	FF61 21A4$_H$	3	32
PORT	Port Control Register	PCR6_10	0000 0010$_H$	32	FF61 21A8$_H$	3	32
PORT	Port Control Register	PCR6_11	0000 0010$_H$	32	FF61 21AC$_H$	3	32
PORT	Port Control Register	PCR7_0	0000 0010$_H$	32	FF61 21C0$_H$	3	32
PORT	Port Control Register	PCR7_1	0000 0010$_H$	32	FF61 21C4$_H$	3	32
PORT	Port Control Register	PCR7_2	0000 0010$_H$	32	FF61 21C8$_H$	3	32
PORT	Port Control Register	PCR7_3	0000 0010$_H$	32	FF61 21CC$_H$	3	32
PORT	Port Control Register	PCR7_4	0000 0010$_H$	32	FF61 21D0$_H$	3	32
PORT	Port Control Register	PCR7_5	0000 0010$_H$	32	FF61 21D4$_H$	3	32
PORT	Port Control Register	PCR7_6	0000 0010$_H$	32	FF61 21D8$_H$	3	32
PORT	Port Control Register	PCR7_7	0000 0010$_H$	32	FF61 21DC$_H$	3	32
PORT	Port Control Register	PCR7_8	0000 0010$_H$	32	FF61 21E0$_H$	3	32
PORT	Port Control Register	PCR7_9	0000 0010$_H$	32	FF61 21E4$_H$	3	32
PORT	Port Control Register	PCR7_10	0000 0010$_H$	32	FF61 21E8$_H$	3	32
PORT	Port Control Register	PCR7_11	0000 0010$_H$	32	FF61 21EC$_H$	3	32
PORT	Port Control Register	PCR8_0	0000 0010$_H$	32	FF61 2200$_H$	3	32
PORT	Port Control Register	PCR8_1	0000 0010$_H$	32	FF61 2204$_H$	3	32
PORT	Port Control Register	PCR8_2	0000 0010$_H$	32	FF61 2208$_H$	3	32
PORT	Port Control Register	PCR8_3	0000 0010$_H$	32	FF61 220C$_H$	3	32
PORT	Port Control Register	PCR8_4	0000 0010$_H$	32	FF61 2210$_H$	3	32
PORT	Port Control Register	PCR8_5	0000 0010$_H$	32	FF61 2214$_H$	3	32
PORT	Port Control Register	PCR8_6	0000 0010$_H$	32	FF61 2218$_H$	3	32

(13/201)

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
PORT	Port Control Register	PCR8_7	0000 0010$_H$	32	FF61 221C$_H$	3	32
PORT	Port Control Register	PCR10_0	0000 0010$_H$	32	FF61 2280$_H$	3	32
PORT	Port Control Register	PCR10_1	0000 0010$_H$	32	FF61 2284$_H$	3	32
PORT	Port Control Register	PCR10_2	0000 0010$_H$	32	FF61 2288$_H$	3	32
PORT	Port Control Register	PCR10_3	0000 0010$_H$	32	FF61 228C$_H$	3	32
PORT	Port Control Register	PCR10_4	0000 0010$_H$	32	FF61 2290$_H$	3	32
PORT	Port Control Register	PCR11_0	0000 0010$_H$	32	FF61 22C0$_H$	3	32
PORT	Port Control Register	PCR11_1	0000 0010$_H$	32	FF61 22C4$_H$	3	32
PORT	Port Control Register	PCR11_2	0000 0010$_H$	32	FF61 22C8$_H$	3	32
PORT	Port Control Register	PCR11_3	0000 0010$_H$	32	FF61 22CC$_H$	3	32
PORT	Port Control Register	PCR11_4	0000 0010$_H$	32	FF61 22D0$_H$	3	32
PORT	Port Control Register	PCR11_5	0000 0010$_H$	32	FF61 22D4$_H$	3	32
PORT	Port Control Register	PCR13_0	0000 0010$_H$	32	FF61 2340$_H$	3	32
PORT	Port Control Register	PCR13_1	0000 0010$_H$	32	FF61 2344$_H$	3	32
PORT	Port Control Register	PCR13_2	0000 0010$_H$	32	FF61 2348$_H$	3	32
PORT	Port Control Register	PCR13_3	0000 0010$_H$	32	FF61 234C$_H$	3	32
PORT	Port Control Register	PCR13_4	0000 0010$_H$	32	FF61 2350$_H$	3	32
PORT	Port Control Register	PCR13_5	0000 0010$_H$	32	FF61 2354$_H$	3	32
PORT	Port Control Register	PCR13_6	0000 0010$_H$	32	FF61 2358$_H$	3	32
PORT	Port Control Register	PCR13_7	0000 0010$_H$	32	FF61 235C$_H$	3	32
PORT	Port Control Register	PCR13_8	0000 0010$_H$	32	FF61 2360$_H$	3	32
PORT	Port Control Register	PCR13_9	0000 0010$_H$	32	FF61 2364$_H$	3	32
PORT	Port Control Register	PCR14_0	0000 0000$_H$	32	FF61 2380$_H$	3	32
PORT	Port Control Register	PCR14_1	0000 0000$_H$	32	FF61 2384$_H$	3	32
PORT	Port Control Register	PCR14_2	0000 0000$_H$	32	FF61 2388$_H$	3	32
PORT	Port Control Register	PCR14_3	0000 0000$_H$	32	FF61 238C$_H$	3	32
PORT	Port Control Register	PCR14_4	0000 0000$_H$	32	FF61 2390$_H$	3	32
PORT	Port Control Register	PCR14_5	0000 0000$_H$	32	FF61 2394$_H$	3	32
PORT	Port Control Register	PCR14_6	0000 0000$_H$	32	FF61 2398$_H$	3	32
PORT	Port Control Register	PCR14_7	0000 0000$_H$	32	FF61 239C$_H$	3	32
PORT	Port Control Register	PCR14_8	0000 0000$_H$	32	FF61 23A0$_H$	3	32
PORT	Port Control Register	PCR14_9	0000 0000$_H$	32	FF61 23A4$_H$	3	32
PORT	Port Control Register	PCR14_10	0000 0000$_H$	32	FF61 23A8$_H$	3	32
PORT	Port Control Register	PCR14_11	0000 0000$_H$	32	FF61 23AC$_H$	3	32
PORT	Port Control Register	PCR14_12	0000 0000$_H$	32	FF61 23B0$_H$	3	32
PORT	Port Control Register	PCR14_13	0000 0000$_H$	32	FF61 23B4$_H$	3	32
PORT	Port Control Register	PCR14_14	0000 0000$_H$	32	FF61 23B8$_H$	3	32
PORT	Port Control Register	PCR14_15	0000 0000$_H$	32	FF61 23BC$_H$	3	32
PORT	Port Control Register	PCR15_0	0000 0000$_H$	32	FF61 23C0$_H$	3	32
PORT	Port Control Register	PCR15_1	0000 0000$_H$	32	FF61 23C4$_H$	3	32
PORT	Port Control Register	PCR15_2	0000 0000$_H$	32	FF61 23C8$_H$	3	32
PORT	Port Control Register	PCR15_3	0000 0000$_H$	32	FF61 23CC$_H$	3	32
PORT	Port Control Register	PCR16_0	0000 0000$_H$	32	FF61 2400$_H$	3	32
PORT	Port Control Register	PCR16_1	0000 0000$_H$	32	FF61 2404$_H$	3	32
PORT	Port Control Register	PCR16_2	0000 0000$_H$	32	FF61 2408$_H$	3	32
PORT	Port Control Register	PCR16_3	0000 0000$_H$	32	FF61 240C$_H$	3	32
PORT	Port Control Register	PCR16_4	0000 0000$_H$	32	FF61 2410$_H$	3	32

(14/201)

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
PORT	Port Control Register	PCR16_5	0000 0000$_H$	32	FF61 2414$_H$	3	32
PORT	Port Control Register	PCR16_6	0000 0000$_H$	32	FF61 2418$_H$	3	32
PORT	Port Control Register	PCR16_7	0000 0000$_H$	32	FF61 241C$_H$	3	32
PORT	Port Control Register	PCR16_8	0000 0000$_H$	32	FF61 2420$_H$	3	32
PORT	Port Control Register	PCR17_0	0000 0010$_H$	32	FF61 2440$_H$	3	32
PORT	Port Control Register	PCR17_1	0000 0010$_H$	32	FF61 2444$_H$	3	32
PORT	Port Control Register	PCR17_2	0000 0010$_H$	32	FF61 2448$_H$	3	32
PORT	Port Control Register	PCR17_3	0000 0010$_H$	32	FF61 244C$_H$	3	32
DFE	Control Register A (Channel 0)	CTLACH0	0000 0000$_H$	32	FFBF 0000$_H$	3	8, 16, 32
DFE	Control Register A (Channel 1)	CTLACH1	0000 0000$_H$	32	FFBF 0004$_H$	3	8, 16, 32
DFE	Control Register A (Channel 2)	CTLACH2	0000 0000$_H$	32	FFBF 0008$_H$	3	8, 16, 32
DFE	Control Register A (Channel 3)	CTLACH3	0000 0000$_H$	32	FFBF 000C$_H$	3	8, 16, 32
DFE	Control Register A (Channel 4)	CTLACH4	0000 0000$_H$	32	FFBF 0010$_H$	3	8, 16, 32
DFE	Control Register A (Channel 5)	CTLACH5	0000 0000$_H$	32	FFBF 0014$_H$	3	8, 16, 32
DFE	Control Register A (Channel 6)	CTLACH6	0000 0000$_H$	32	FFBF 0018$_H$	3	8, 16, 32
DFE	Control Register A (Channel 7)	CTLACH7	0000 0000$_H$	32	FFBF 001C$_H$	3	8, 16, 32
DFE	Control Register A (Channel 8)	CTLACH8	0000 0000$_H$	32	FFBF 0020$_H$	3	8, 16, 32
DFE	Control Register A (Channel 9)	CTLACH9	0000 0000$_H$	32	FFBF 0024$_H$	3	8, 16, 32
DFE	Control Register A (Channel 10)	CTLACH10	0000 0000$_H$	32	FFBF 0028$_H$	3	8, 16, 32
DFE	Control Register A (Channel 11)	CTLACH11	0000 0000$_H$	32	FFBF 002C$_H$	3	8, 16, 32
DFE	Control Register A (Channel 12)	CTLACH12	0000 0000$_H$	32	FFBF 0030$_H$	3	8, 16, 32
DFE	Control Register A (Channel 13)	CTLACH13	0000 0000$_H$	32	FFBF 0034$_H$	3	8, 16, 32
DFE	Control Register A (Channel 14)	CTLACH14	0000 0000$_H$	32	FFBF 0038$_H$	3	8, 16, 32
DFE	Control Register A (Channel 15)	CTLACH15	0000 0000$_H$	32	FFBF 003C$_H$	3	8, 16, 32
DFE	Control Register B (Channel 0)	CTLBCH0	0000 0000$_H$	32	FFBF 0040$_H$	3	8, 16, 32
DFE	Control Register B (Channel 1)	CTLBCH1	0000 0000$_H$	32	FFBF 0044$_H$	3	8, 16, 32
DFE	Control Register B (Channel 2)	CTLBCH2	0000 0000$_H$	32	FFBF 0048$_H$	3	8, 16, 32
DFE	Control Register B (Channel 3)	CTLBCH3	0000 0000$_H$	32	FFBF 004C$_H$	3	8, 16, 32
DFE	Control Register B (Channel 4)	CTLBCH4	0000 0000$_H$	32	FFBF 0050$_H$	3	8, 16, 32
DFE	Control Register B (Channel 5)	CTLBCH5	0000 0000$_H$	32	FFBF 0054$_H$	3	8, 16, 32
DFE	Control Register B (Channel 6)	CTLBCH6	0000 0000$_H$	32	FFBF 0058$_H$	3	8, 16, 32
DFE	Control Register B (Channel 7)	CTLBCH7	0000 0000$_H$	32	FFBF 005C$_H$	3	8, 16, 32
DFE	Control Register B (Channel 8)	CTLBCH8	0000 0000$_H$	32	FFBF 0060$_H$	3	8, 16, 32
DFE	Control Register B (Channel 9)	CTLBCH9	0000 0000$_H$	32	FFBF 0064$_H$	3	8, 16, 32
DFE	Control Register B (Channel 10)	CTLBCH10	0000 0000$_H$	32	FFBF 0068$_H$	3	8, 16, 32
DFE	Control Register B (Channel 11)	CTLBCH11	0000 0000$_H$	32	FFBF 006C$_H$	3	8, 16, 32
DFE	Control Register B (Channel 12)	CTLBCH12	0000 0000$_H$	32	FFBF 0070$_H$	3	8, 16, 32
DFE	Control Register B (Channel 13)	CTLBCH13	0000 0000$_H$	32	FFBF 0074$_H$	3	8, 16, 32
DFE	Control Register B (Channel 14)	CTLBCH14	0000 0000$_H$	32	FFBF 0078$_H$	3	8, 16, 32
DFE	Control Register B (Channel 15)	CTLBCH15	0000 0000$_H$	32	FFBF 007C$_H$	3	8, 16, 32
DFE	Output Data Register (Channel 0)	DOCH0	0000 0000$_H$	32	FFBF 0080$_H$	3	32
DFE	Output Data Register (Channel 1)	DOCH1	0000 0000$_H$	32	FFBF 0084$_H$	3	32
DFE	Output Data Register (Channel 2)	DOCH2	0000 0000$_H$	32	FFBF 0088$_H$	3	32
DFE	Output Data Register (Channel 3)	DOCH3	0000 0000$_H$	32	FFBF 008C$_H$	3	32
DFE	Output Data Register (Channel 4)	DOCH4	0000 0000$_H$	32	FFBF 0090$_H$	3	32
DFE	Output Data Register (Channel 5)	DOCH5	0000 0000$_H$	32	FFBF 0094$_H$	3	32
DFE	Output Data Register (Channel 6)	DOCH6	0000 0000$_H$	32	FFBF 0098$_H$	3	32

(15/201)

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
DFE	Output Data Register (Channel 7)	DOCH7	0000 0000$_H$	32	FFBF 009C$_H$	3	32
DFE	Output Data Register (Channel 8)	DOCH8	0000 0000$_H$	32	FFBF 00A0$_H$	3	32
DFE	Output Data Register (Channel 9)	DOCH9	0000 0000$_H$	32	FFBF 00A4$_H$	3	32
DFE	Output Data Register (Channel 10)	DOCH10	0000 0000$_H$	32	FFBF 00A8$_H$	3	32
DFE	Output Data Register (Channel 11)	DOCH11	0000 0000$_H$	32	FFBF 00AC$_H$	3	32
DFE	Output Data Register (Channel 12)	DOCH12	0000 0000$_H$	32	FFBF 00B0$_H$	3	32
DFE	Output Data Register (Channel 13)	DOCH13	0000 0000$_H$	32	FFBF 00B4$_H$	3	32
DFE	Output Data Register (Channel 14)	DOCH14	0000 0000$_H$	32	FFBF 00B8$_H$	3	32
DFE	Output Data Register (Channel 15)	DOCH15	0000 0000$_H$	32	FFBF 00BC$_H$	3	32
DFE	PH Result Register (Channel 0)	PHCH0	8000 0000$_H$	32	FFBF 00C0$_H$	3	32
DFE	PH Result Register (Channel 1)	PHCH1	8000 0000$_H$	32	FFBF 00C4$_H$	3	32
DFE	PH Result Register (Channel 2)	PHCH2	8000 0000$_H$	32	FFBF 00C8$_H$	3	32
DFE	PH Result Register (Channel 3)	PHCH3	8000 0000$_H$	32	FFBF 00CC$_H$	3	32
DFE	PH Result Register (Channel 4)	PHCH4	8000 0000$_H$	32	FFBF 00D0$_H$	3	32
DFE	PH Result Register (Channel 5)	PHCH5	8000 0000$_H$	32	FFBF 00D4$_H$	3	32
DFE	PH Result Register (Channel 6)	PHCH6	8000 0000$_H$	32	FFBF 00D8$_H$	3	32
DFE	PH Result Register (Channel 7)	PHCH7	8000 0000$_H$	32	FFBF 00DC$_H$	3	32
DFE	PH Result Register (Channel 8)	PHCH8	8000 0000$_H$	32	FFBF 00E0$_H$	3	32
DFE	PH Result Register (Channel 9)	PHCH9	8000 0000$_H$	32	FFBF 00E4$_H$	3	32
DFE	Status Register (Channel 0)	STCH0	0000$_H$	16	FFBF 0140$_H$	3	8, 16
DFE	Status Register (Channel 1)	STCH1	0000$_H$	16	FFBF 0144$_H$	3	8, 16
DFE	Status Register (Channel 2)	STCH2	0000$_H$	16	FFBF 0148$_H$	3	8, 16
DFE	Status Register (Channel 3)	STCH3	0000$_H$	16	FFBF 014C$_H$	3	8, 16
DFE	Status Register (Channel 4)	STCH4	0000$_H$	16	FFBF 0150$_H$	3	8, 16
DFE	Status Register (Channel 5)	STCH5	0000$_H$	16	FFBF 0154$_H$	3	8, 16
DFE	Status Register (Channel 6)	STCH6	0000$_H$	16	FFBF 0158$_H$	3	8, 16
DFE	Status Register (Channel 7)	STCH7	0000$_H$	16	FFBF 015C$_H$	3	8, 16
DFE	Status Register (Channel 8)	STCH8	0000$_H$	16	FFBF 0160$_H$	3	8, 16
DFE	Status Register (Channel 9)	STCH9	0000$_H$	16	FFBF 0164$_H$	3	8, 16
DFE	Status Register (Channel 10)	STCH10	0000$_H$	16	FFBF 0168$_H$	3	8, 16
DFE	Status Register (Channel 11)	STCH11	0000$_H$	16	FFBF 016C$_H$	3	8, 16
DFE	Status Register (Channel 12)	STCH12	0000$_H$	16	FFBF 0170$_H$	3	8, 16
DFE	Status Register (Channel 13)	STCH13	0000$_H$	16	FFBF 0174$_H$	3	8, 16
DFE	Status Register (Channel 14)	STCH14	0000$_H$	16	FFBF 0178$_H$	3	8, 16
DFE	Status Register (Channel 15)	STCH15	0000$_H$	16	FFBF 017C$_H$	3	8, 16
DFE	Clear Status Register (Channel 0)	CLRSTCH0	0000$_H$	16	FFBF 0180$_H$	3	8, 16
DFE	Clear Status Register (Channel 1)	CLRSTCH1	0000$_H$	16	FFBF 0184$_H$	3	8, 16
DFE	Clear Status Register (Channel 2)	CLRSTCH2	0000$_H$	16	FFBF 0188$_H$	3	8, 16
DFE	Clear Status Register (Channel 3)	CLRSTCH3	0000$_H$	16	FFBF 018C$_H$	3	8, 16
DFE	Clear Status Register (Channel 4)	CLRSTCH4	0000$_H$	16	FFBF 0190$_H$	3	8, 16
DFE	Clear Status Register (Channel 5)	CLRSTCH5	0000$_H$	16	FFBF 0194$_H$	3	8, 16
DFE	Clear Status Register (Channel 6)	CLRSTCH6	0000$_H$	16	FFBF 0198$_H$	3	8, 16
DFE	Clear Status Register (Channel 7)	CLRSTCH7	0000$_H$	16	FFBF 019C$_H$	3	8, 16
DFE	Clear Status Register (Channel 8)	CLRSTCH8	0000$_H$	16	FFBF 01A0$_H$	3	8, 16
DFE	Clear Status Register (Channel 9)	CLRSTCH9	0000$_H$	16	FFBF 01A4$_H$	3	8, 16
DFE	Clear Status Register (Channel 10)	CLRSTCH10	0000$_H$	16	FFBF 01A8$_H$	3	8, 16
DFE	Clear Status Register (Channel 11)	CLRSTCH11	0000$_H$	16	FFBF 01AC$_H$	3	8, 16

(16/201)

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
DFE	Clear Status Register (Channel 12)	CLRSTCH12	0000_H	16	FFBF $01B0_H$	3	8, 16
DFE	Clear Status Register (Channel 13)	CLRSTCH13	0000_H	16	FFBF $01B4_H$	3	8, 16
DFE	Clear Status Register (Channel 14)	CLRSTCH14	0000_H	16	FFBF $01B8_H$	3	8, 16
DFE	Clear Status Register (Channel 15)	CLRSTCH15	0000_H	16	FFBF $01BC_H$	3	8, 16
DFE	Error Mask Register (Channel 0)	ERMCH0	00_H	8	FFBF $01C0_H$	3	8
DFE	Error Mask Register (Channel 1)	ERMCH1	00_H	8	FFBF $01C4_H$	3	8
DFE	Error Mask Register (Channel 2)	ERMCH2	00_H	8	FFBF $01C8_H$	3	8
DFE	Error Mask Register (Channel 3)	ERMCH3	00_H	8	FFBF $01CC_H$	3	8
DFE	Error Mask Register (Channel 4)	ERMCH4	00_H	8	FFBF $01D0_H$	3	8
DFE	Error Mask Register (Channel 5)	ERMCH5	00_H	8	FFBF $01D4_H$	3	8
DFE	Error Mask Register (Channel 6)	ERMCH6	00_H	8	FFBF $01D8_H$	3	8
DFE	Error Mask Register (Channel 7)	ERMCH7	00_H	8	FFBF $01DC_H$	3	8
DFE	Error Mask Register (Channel 8)	ERMCH8	00_H	8	FFBF $01E0_H$	3	8
DFE	Error Mask Register (Channel 9)	ERMCH9	00_H	8	FFBF $01E4_H$	3	8
DFE	Error Mask Register (Channel 10)	ERMCH10	00_H	8	FFBF $01E8_H$	3	8
DFE	Error Mask Register (Channel 11)	ERMCH11	00_H	8	FFBF $01EC_H$	3	8
DFE	Error Mask Register (Channel 12)	ERMCH12	00_H	8	FFBF $01F0_H$	3	8
DFE	Error Mask Register (Channel 13)	ERMCH13	00_H	8	FFBF $01F4_H$	3	8
DFE	Error Mask Register (Channel 14)	ERMCH14	00_H	8	FFBF $01F8_H$	3	8
DFE	Error Mask Register (Channel 15)	ERMCH15	00_H	8	FFBF $01FC_H$	3	8
DFE	Trigger Setting Register (Channel 0)	TRGCH0	0000 0000_H	32	FFBF 0200_H	3	16, 32
DFE	Trigger Setting Register (Channel 1)	TRGCH1	0000 0000_H	32	FFBF 0204_H	3	16, 32
DFE	Trigger Setting Register (Channel 2)	TRGCH2	0000 0000_H	32	FFBF 0208_H	3	16, 32
DFE	Trigger Setting Register (Channel 3)	TRGCH3	0000 0000_H	32	FFBF $020C_H$	3	16, 32
DFE	Trigger Setting Register (Channel 4)	TRGCH4	0000 0000_H	32	FFBF 0210_H	3	16, 32
DFE	Trigger Setting Register (Channel 5)	TRGCH5	0000 0000_H	32	FFBF 0214_H	3	16, 32
DFE	Trigger Setting Register (Channel 6)	TRGCH6	0000 0000_H	32	FFBF 0218_H	3	16, 32
DFE	Trigger Setting Register (Channel 7)	TRGCH7	0000 0000_H	32	FFBF $021C_H$	3	16, 32
DFE	Trigger Setting Register (Channel 8)	TRGCH8	0000 0000_H	32	FFBF 0220_H	3	16, 32
DFE	Trigger Setting Register (Channel 9)	TRGCH9	0000 0000_H	32	FFBF 0224_H	3	16, 32
DFE	Trigger Setting Register (Channel 10)	TRGCH10	0000 0000_H	32	FFBF 0228_H	3	16, 32
DFE	Trigger Setting Register (Channel 11)	TRGCH11	0000 0000_H	32	FFBF $022C_H$	3	16, 32
DFE	Trigger Setting Register (Channel 12)	TRGCH12	0000 0000_H	32	FFBF 0230_H	3	16, 32
DFE	Trigger Setting Register (Channel 13)	TRGCH13	0000 0000_H	32	FFBF 0234_H	3	16, 32
DFE	Trigger Setting Register (Channel 14)	TRGCH14	0000 0000_H	32	FFBF 0238_H	3	16, 32
DFE	Trigger Setting Register (Channel 15)	TRGCH15	0000 0000_H	32	FFBF $023C_H$	3	16, 32
DFE	Trigger History Register (Channel 0)	TRHCH0	00_H	8	FFBF 0240_H	3	8
DFE	Trigger History Register (Channel 1)	TRHCH1	00_H	8	FFBF 0244_H	3	8
DFE	Trigger History Register (Channel 2)	TRHCH2	00_H	8	FFBF 0248_H	3	8
DFE	Trigger History Register (Channel 3)	TRHCH3	00_H	8	FFBF $024C_H$	3	8
DFE	Trigger History Register (Channel 4)	TRHCH4	00_H	8	FFBF 0250_H	3	8
DFE	Trigger History Register (Channel 5)	TRHCH5	00_H	8	FFBF 0254_H	3	8
DFE	Trigger History Register (Channel 6)	TRHCH6	00_H	8	FFBF 0258_H	3	8
DFE	Trigger History Register (Channel 7)	TRHCH7	00_H	8	FFBF $025C_H$	3	8
DFE	Trigger History Register (Channel 8)	TRHCH8	00_H	8	FFBF 0260_H	3	8
DFE	Trigger History Register (Channel 9)	TRHCH9	00_H	8	FFBF 0264_H	3	8
DFE	Trigger History Register (Channel 10)	TRHCH10	00_H	8	FFBF 0268_H	3	8

(17/201)

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
DFE	Trigger History Register (Channel 11)	TRHCH11	00_H	8	FFBF 026C$_H$	3	8
DFE	Trigger History Register (Channel 12)	TRHCH12	00_H	8	FFBF 0270$_H$	3	8
DFE	Trigger History Register (Channel 13)	TRHCH13	00_H	8	FFBF 0274$_H$	3	8
DFE	Trigger History Register (Channel 14)	TRHCH14	00_H	8	FFBF 0278$_H$	3	8
DFE	Trigger History Register (Channel 15)	TRHCH15	00_H	8	FFBF 027C$_H$	3	8
DFE	Comparison Value Setting Register A	CPA	0000 0000_H	32	FFBF 0280$_H$	3	32
DFE	Comparison Value Setting Register B	CPB	0000 0000_H	32	FFBF 0284$_H$	3	32
DFE	Comparison Value Setting Register C	CPC	0000 0000_H	32	FFBF 0288$_H$	3	32
DFE	Comparison Value Setting Register D	CPD	0000 0000_H	32	FFBF 028C$_H$	3	32
DFE	PH Initial Value Setting Register A	PHIA	8000 0000_H	32	FFBF 0290$_H$	3	32
DFE	PH Initial Value Setting Register B	PHIB	8000 0000_H	32	FFBF 0294$_H$	3	32
DFE	PH Initial Value Setting Register C	PHIC	8000 0000_H	32	FFBF 0298$_H$	3	32
DFE	PH Initial Value Setting Register D	PHID	8000 0000_H	32	FFBF 029C$_H$	3	32
DFE	Accumulation / Decimation Count Setting Register A	ACA	0000_H	16	FFBF 02B0$_H$	3	16
DFE	Accumulation / Decimation Count Setting Register B	ACB	0000_H	16	FFBF 02B4$_H$	3	16
DFE	Accumulation / Decimation Count Setting Register C	ACC	0000_H	16	FFBF 02B8$_H$	3	16
DFE	Accumulation / Decimation Count Setting Register D	ACD	0000_H	16	FFBF 02BC$_H$	3	16
DFE	Software Input Data Register	DI	0000 0000_H	32	FFBF 02C0$_H$	3	32
DFE	Software Trigger Register	TRG	00_H	8	FFBF 02C4$_H$	3	8
DFE	DFE Status Register	ST	0000 0000_H	32	FFBF 02CC$_H$	3	8, 16, 32
DFE	PH Initialization / End Timer Trigger Select Register	PITRG	0000 0000_H	32	FFBF 0300$_H$	3	8, 16, 32
DFE	Accumulation/Decimation Initialization/ Prohibition Timer Trigger Select Register	MITRG	0000 0000_H	32	FFBF 0304$_H$	3	8, 16, 32
DFE	Filter Initialization Timer Trigger Select Register	FITRG	0000 0000_H	32	FFBF 0308$_H$	3	8, 16, 32
DFE	PH Update Notification Setting Register 0	PHUPDC0	00_H	8	FFBF 0310$_H$	3	8
DFE	PH Update Notification Setting Register 1	PHUPDC1	00_H	8	FFBF 0314$_H$	3	8
EINT	NMI Interrupt Control Register	NMICTL	00_H	8	FFC0 0000_H	3	8
EINT	External Interrupt Control Register	EXINTCTL	0000_H	16	FFC0 0010_H	3	8, 16
EINT	External Interrupt Status Register	EXINTSTR	00_H	8	FFC0 0014$_H$	3	8
EINT	External Interrupt Status Clear Register	EXINTSTC	00_H	8	FFC0 0018$_H$	3	8
EINT	Software Interrupt Register 0	SINTR0	00_H	8	FFC0 0020$_H$	3	8
EINT	Software Interrupt Register 1	SINTR1	00_H	8	FFC0 0024$_H$	3	8
EINT	Software Interrupt Register 2	SINTR2	00_H	8	FFC0 0028$_H$	3	8
EINT	Software Interrupt Register 3	SINTR3	00_H	8	FFC0 002C$_H$	3	8
ACK0	Clock Control Register	ACK0CKC	01_H	8	FFC0 4000$_H$	3	8
ACK0	CKC Flag Register	ACK0CKCFLG	01_H	8	FFC0 4004$_H$	3	8
ACK0	Clock Selection Register	ACK0CKCTL	00_H	8	FFC0 4008$_H$	3	8
ACK0	BRGA0 Comparison Register	ACK0BRGA0CMP	00_H	8	FFC0 400C$_H$	3	8
ACK0	CKC Protection Command Register	ACK0CKCPCMD	Undefined	8	FFC0 4100$_H$	3	8
ACK0	CKC Protection Status Register	ACK0CKCPS	00_H	8	FFC0 4104$_H$	3	8
MSTB	Module Standby Register -TSG2	MSRTSG	00_H	8	FFC0 5000$_H$	3	8
MSTB	Module Standby Register -TAPA	MSRTAPA	00_H	8	FFC0 5004$_H$	3	8
MSTB	Module Standby Register -OSTM	MSROSTM	00_H	8	FFC0 5008$_H$	3	8
MSTB	Module Standby Register -WDTA	MSRWDTA	00_H	8	FFC0 500C$_H$	3	8

(18/201)

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
MSTB	Module Standby Register -PIC	MSRPIC	00_H	8	FFC0 5010_H	3	8
MSTB	Module Standby Register -RS-CAN	MSRRCAN	00_H	8	FFC0 5014_H	3	8
MSTB	Module Standby Register -FlexRay	MSRFRAY	00_H	8	FFC0 5018_H	3	8
MSTB	Module Standby Register -RLIN2	MSRRLIN	00_H	8	FFC0 $501C_H$	3	8
MSTB	Module Standby Register -SCI3	MSRSCI	00_H	8	FFC0 5020_H	3	8
MSTB	Module Standby Register -CSIH	MSRCSIH	00_H	8	FFC0 5024_H	3	8
MSTB	Module Standby Register -SAR AD	MSRSAD	00_H	8	FFC0 5028_H	3	8
MSTB	Module Standby Register -ΔΣAD	MSRDAD	00_H	8	FFC0 $502C_H$	3	8
MSTB	Module Standby Register -ATU-IV	MSRATU	00_H	8	FFC0 5030_H	3	8
MSTB	Module Standby Register -APA	MSRAPA	00_H	8	FFC0 5034_H	3	8
MSTB	Module Standby Register -DFE	MSRDFE	00_H	8	FFC0 5038_H	3	8
MSTB	Module Standby Register -RHSB	MSRRHSB	00_H	8	FFC0 $503C_H$	3	8
MSTB	MSR Protection Command Register	MSRPCMD	Undefined	8	FFC0 5070_H	3	8
MSTB	MSR Protection Status Register	MSRPS	00_H	8	FFC0 5074_H	3	8
PORT	Port Input Buffer Control Register	PIBC0	0000_H	16	FFC1 4000_H	3	16
PORT	Port Bidirectional Control Register	PBDC0	0000_H	16	FFC1 4004_H	3	16
PORT	Port IP Control Register	PIPC0	0000_H	16	FFC1 4008_H	3	16
PORT	Pull-up Option Register	PU0	0000_H	16	FFC1 $400C_H$	3	16
PORT	Pull-down Option Register	PD0	0000_H	16	FFC1 4010_H	3	16
PORT	Port Drive Strength Control Register	PDSC0	0000 0000_H	32	FFC1 4018_H	3	32
PORT	Port Universal Control Register	PUCC0	0000 0000_H	32	FFC1 4028_H	3	32
PORT	Port Register Protection Command Register	PPCMD0	00_H	8	FFC1 $402C_H$	3	8
PORT	Port Protection Status Register	PPROTS0	00_H	8	FFC1 4034_H	3	8
PORT	Port Input Buffer Control Register	PIBC1	0000_H	16	FFC1 4040_H	3	16
PORT	Port Bidirectional Control Register	PBDC1	0000_H	16	FFC1 4044_H	3	16
PORT	Port IP Control Register	PIPC1	0000_H	16	FFC1 4048_H	3	16
PORT	Pull-up Option Register	PU1	0000_H	16	FFC1 $404C_H$	3	16
PORT	Pull-down Option Register	PD1	0000_H	16	FFC1 4050_H	3	16
PORT	Port Drive Strength Control Register	PDSC1	0000 0000_H	32	FFC1 4058_H	3	32
PORT	Port Universal Control Register	PUCC1	0000 0000_H	32	FFC1 4068_H	3	32
PORT	Port Register Protection Command Register	PPCMD1	00_H	8	FFC1 $406C_H$	3	8
PORT	Port Protection Status Register	PPROTS1	00_H	8	FFC1 4074_H	3	8
PORT	Port Input Buffer Control Register	PIBC2	0000_H	16	FFC1 4080_H	3	16
PORT	Port Bidirectional Control Register	PBDC2	0000_H	16	FFC1 4084_H	3	16
PORT	Port IP Control Register	PIPC2	0000_H	16	FFC1 4088_H	3	16
PORT	Pull-up Option Register	PU2	0000_H	16	FFC1 $408C_H$	3	16
PORT	Pull-down Option Register	PD2	0000_H	16	FFC1 4090_H	3	16
PORT	Port Drive Strength Control Register	PDSC2	0000 0000_H	32	FFC1 4098_H	3	32
PORT	Port Universal Control Register	PUCC2	0000 0000_H	32	FFC1 $40A8_H$	3	32
PORT	Port Register Protection Command Register	PPCMD2	00_H	8	FFC1 $40AC_H$	3	8
PORT	Port Protection Status Register	PPROTS2	00_H	8	FFC1 $40B4_H$	3	8
PORT	Port Input Buffer Control Register	PIBC3	0000_H	16	FFC1 $40C0_H$	3	16
PORT	Port Bidirectional Control Register	PBDC3	0000_H	16	FFC1 $40C4_H$	3	16
PORT	Port IP Control Register	PIPC3	0000_H	16	FFC1 $40C8_H$	3	16
PORT	Pull-up Option Register	PU3	0000_H	16	FFC1 $40CC_H$	3	16

(19/201)

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
PORT	Pull-down Option Register	PD3	0000_H	16	FFC1 40D0$_H$	3	16
PORT	Port Drive Strength Control Register	PDSC3	0000 0000_H	32	FFC1 40D8$_H$	3	32
PORT	Port Universal Control Register	PUCC3	0000 0000_H	32	FFC1 40E8$_H$	3	32
PORT	Port Register Protection Command Register	PPCMD3	00_H	8	FFC1 40EC$_H$	3	8
PORT	Port Protection Status Register	PPROTS3	00_H	8	FFC1 40F4$_H$	3	8
PORT	Port Input Buffer Control Register	PIBC4	0000_H	16	FFC1 4100$_H$	3	16
PORT	Port Bidirectional Control Register	PBDC4	0000_H	16	FFC1 4104$_H$	3	16
PORT	Port IP Control Register	PIPC4	0000_H	16	FFC1 4108$_H$	3	16
PORT	Pull-up Option Register	PU4	0000_H	16	FFC1 410C$_H$	3	16
PORT	Pull-down Option Register	PD4	0000_H	16	FFC1 4110$_H$	3	16
PORT	Port Drive Strength Control Register	PDSC4	0000 0000_H	32	FFC1 4118$_H$	3	32
PORT	Port Universal Control Register	PUCC4	0000 0000_H	32	FFC1 4128$_H$	3	32
PORT	Port Register Protection Command Register	PPCMD4	00_H	8	FFC1 412C$_H$	3	8
PORT	Port Protection Status Register	PPROTS4	00_H	8	FFC1 4134$_H$	3	8
PORT	Port Input Buffer Control Register	PIBC5	0000_H	16	FFC1 4140$_H$	3	16
PORT	Port Bidirectional Control Register	PBDC5	0000_H	16	FFC1 4144$_H$	3	16
PORT	Port IP Control Register	PIPC5	0000_H	16	FFC1 4148$_H$	3	16
PORT	Pull-up Option Register	PU5	0000_H	16	FFC1 414C$_H$	3	16
PORT	Pull-down Option Register	PD5	0000_H	16	FFC1 4150$_H$	3	16
PORT	Port Drive Strength Control Register	PDSC5	0000 0000_H	32	FFC1 4158$_H$	3	32
PORT	Port Universal Control Register	PUCC5	0000 0000_H	32	FFC1 4168$_H$	3	32
PORT	Port Register Protection Command Register	PPCMD5	00_H	8	FFC1 416C$_H$	3	8
PORT	Port Protection Status Register	PPROTS5	00_H	8	FFC1 4174$_H$	3	8
PORT	Port Input Buffer Control Register	PIBC6	0000_H	16	FFC1 4180$_H$	3	16
PORT	Port Bidirectional Control Register	PBDC6	0000_H	16	FFC1 4184$_H$	3	16
PORT	Port IP Control Register	PIPC6	0000_H	16	FFC1 4188$_H$	3	16
PORT	Pull-up Option Register	PU6	0000_H	16	FFC1 418C$_H$	3	16
PORT	Pull-down Option Register	PD6	0000_H	16	FFC1 4190$_H$	3	16
PORT	Port Drive Strength Control Register	PDSC6	0000 0000_H	32	FFC1 4198$_H$	3	32
PORT	Port Universal Control Register	PUCC6	0000 0000_H	32	FFC1 41A8$_H$	3	32
PORT	Port Register Protection Command Register	PPCMD6	00_H	8	FFC1 41AC$_H$	3	8
PORT	Port Protection Status Register	PPROTS6	00_H	8	FFC1 41B4$_H$	3	8
PORT	Port Input Buffer Control Register	PIBC7	0000_H	16	FFC1 41C0$_H$	3	16
PORT	Port Bidirectional Control Register	PBDC7	0000_H	16	FFC1 41C4$_H$	3	16
PORT	Port IP Control Register	PIPC7	0000_H	16	FFC1 41C8$_H$	3	16
PORT	Pull-up Option Register	PU7	0000_H	16	FFC1 41CC$_H$	3	16
PORT	Pull-down Option Register	PD7	0000_H	16	FFC1 41D0$_H$	3	16
PORT	Port Drive Strength Control Register	PDSC7	0000 0000_H	32	FFC1 41D8$_H$	3	32
PORT	Port Universal Control Register	PUCC7	0000 0000_H	32	FFC1 41E8$_H$	3	32
PORT	Port Register Protection Command Register	PPCMD7	00_H	8	FFC1 41EC$_H$	3	8
PORT	Port Protection Status Register	PPROTS7	00_H	8	FFC1 41F4$_H$	3	8
PORT	Port Input Buffer Control Register	PIBC8	0000_H	16	FFC1 4200$_H$	3	16
PORT	Port Bidirectional Control Register	PBDC8	0000_H	16	FFC1 4204$_H$	3	16
PORT	Port IP Control Register	PIPC8	0000_H	16	FFC1 4208$_H$	3	16

(20/201)

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
PORT	Pull-up Option Register	PU8	0000_H	16	FFC1 420C$_H$	3	16
PORT	Pull-down Option Register	PD8	0000_H	16	FFC1 4210$_H$	3	16
PORT	Port Drive Strength Control Register	PDSC8	0000 0000_H	32	FFC1 4218$_H$	3	32
PORT	Port Universal Control Register	PUCC8	0000 0000_H	32	FFC1 4228$_H$	3	32
PORT	Port Register Protection Command Register	PPCMD8	00_H	8	FFC1 422C$_H$	3	8
PORT	Port Protection Status Register	PPROTS8	00_H	8	FFC1 4234$_H$	3	8
PORT	Port Input Buffer Control Register	PIBC10	0000_H	16	FFC1 4280$_H$	3	16
PORT	Port Bidirectional Control Register	PBDC10	0000_H	16	FFC1 4284$_H$	3	16
PORT	Port IP Control Register	PIPC10	0000_H	16	FFC1 4288$_H$	3	16
PORT	Pull-up Option Register	PU10	0000_H	16	FFC1 428C$_H$	3	16
PORT	Pull-down Option Register	PD10	0000_H	16	FFC1 4290$_H$	3	16
PORT	Port Drive Strength Control Register	PDSC10	0000 0000_H	32	FFC1 4298$_H$	3	32
PORT	Port Universal Control Register	PUCC10	0000 0000_H	32	FFC1 42A8$_H$	3	32
PORT	Port Register Protection Command Register	PPCMD10	00_H	8	FFC1 42AC$_H$	3	8
PORT	Port Protection Status Register	PPROTS10	00_H	8	FFC1 42B4$_H$	3	8
PORT	Port Input Buffer Control Register	PIBC11	0000_H	16	FFC1 42C0$_H$	3	16
PORT	Port Bidirectional Control Register	PBDC11	0000_H	16	FFC1 42C4$_H$	3	16
PORT	Pull-up Option Register	PU11	0000_H	16	FFC1 42CC$_H$	3	16
PORT	Pull-down Option Register	PD11	0000_H	16	FFC1 42D0$_H$	3	16
PORT	Port Drive Strength Control Register	PDSC11	0000 0000_H	32	FFC1 42D8$_H$	3	32
PORT	Port Universal Control Register	PUCC11	0000 0000_H	32	FFC1 42E8$_H$	3	32
PORT	Port Register Protection Command Register	PPCMD11	00_H	8	FFC1 42EC$_H$	3	8
PORT	Port Protection Status Register	PPROTS11	00_H	8	FFC1 42F4$_H$	3	8
PORT	Port Input Buffer Control Register	PIBC13	0000_H	16	FFC1 4340$_H$	3	16
PORT	Port Bidirectional Control Register	PBDC13	0000_H	16	FFC1 4344$_H$	3	16
PORT	Port IP Control Register	PIPC13	0000_H	16	FFC1 4348$_H$	3	16
PORT	Pull-up Option Register	PU13	0000_H	16	FFC1 434C$_H$	3	16
PORT	Pull-down Option Register	PD13	0000_H	16	FFC1 4350$_H$	3	16
PORT	Port Drive Strength Control Register	PDSC13	0000 0000_H	32	FFC1 4358$_H$	3	32
PORT	Port Universal Control Register	PUCC13	0000 0000_H	32	FFC1 4368$_H$	3	32
PORT	Port Register Protection Command Register	PPCMD13	00_H	8	FFC1 436C$_H$	3	8
PORT	Port Protection Status Register	PPROTS13	00_H	8	FFC1 4374$_H$	3	8
PORT	Port Input Buffer Control Register	PIBC14	0000_H	16	FFC1 4380$_H$	3	16
PORT	Port Input Buffer Control Register	PIBC15	0000_H	16	FFC1 43C0$_H$	3	16
PORT	Port Input Buffer Control Register	PIBC16	0000_H	16	FFC1 4400$_H$	3	16
DNF	Digital Noise Elimination Control Register	DNFP010CTL0	00_H	8	FFC3 0000$_H$	3	8
DNF	Digital Noise Elimination Control Register	DNFP010CTL1	00_H	8	FFC3 0004$_H$	3	8
DNF	Digital Noise Elimination Control Register	DNFP010CTL2	00_H	8	FFC3 0008$_H$	3	8
DNF	Digital Noise Elimination Control Register	DNFP010CTL3	00_H	8	FFC3 000C$_H$	3	8
DNF	Digital Noise Elimination Control Register	DNFP010CTL4	00_H	8	FFC3 0010$_H$	3	8
DNF	Digital Noise Elimination Control Register	DNFP010CTL5	00_H	8	FFC3 0014$_H$	3	8

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
DNF	Digital Noise Elimination Control Register	DNFP010CTL6	00_H	8	FFC3 0018_H	3	8
DNF	Digital Noise Elimination Control Register	DNFP010CTL7	00_H	8	FFC3 $001C_H$	3	8
DNF	Digital Noise Elimination Control Register	DNFP010CTL10	00_H	8	FFC3 0028_H	3	8
DNF	Digital Noise Elimination Control Register	DNFP010CTL11	00_H	8	FFC3 $002C_H$	3	8
DNF	Digital Noise Elimination Control Register	DNFP011CTL0	00_H	8	FFC3 0100_H	3	8
DNF	Digital Noise Elimination Control Register	DNFP011CTL1	00_H	8	FFC3 0104_H	3	8
DNF	Digital Noise Elimination Control Register	DNFP011CTL2	00_H	8	FFC3 0108_H	3	8
DNF	Digital Noise Elimination Control Register	DNFP011CTL3	00_H	8	FFC3 $010C_H$	3	8
DNF	Digital Noise Elimination Control Register	DNFP011CTL4	00_H	8	FFC3 0110_H	3	8
DNF	Digital Noise Elimination Control Register	DNFP011CTL5	00_H	8	FFC3 0114_H	3	8
DNF	Digital Noise Elimination Control Register	DNFP011CTL6	00_H	8	FFC3 0118_H	3	8
DNF	Digital Noise Elimination Control Register	DNFP011CTL7	00_H	8	FFC3 $011C_H$	3	8
DNF	Digital Noise Elimination Control Register	DNFP011CTL8	00_H	8	FFC3 0120_H	3	8
DNF	Digital Noise Elimination Control Register	DNFP011CTL9	00_H	8	FFC3 0124_H	3	8
DNF	Digital Noise Elimination Control Register	DNFP022CTL	00_H	8	FFC3 0200_H	3	8
DNF	Edge Detection Control Register	DNFP022EDC0	00_H	8	FFC3 0204_H	3	8
DNF	Edge Detection Flag Register	DNFP022EDF0	00_H	8	FFC3 0208_H	3	8
DNF	Edge Detection Control Register	DNFP022EDC1	00_H	8	FFC3 $020C_H$	3	8
DNF	Edge Detection Flag Register	DNFP022EDF1	00_H	8	FFC3 0210_H	3	8
DNF	Edge Detection Control Register	DNFP022EDC2	00_H	8	FFC3 0214_H	3	8
DNF	Edge Detection Flag Register	DNFP022EDF2	00_H	8	FFC3 0218_H	3	8
DNF	Edge Detection Control Register	DNFP022EDC3	00_H	8	FFC3 $021C_H$	3	8
DNF	Edge Detection Flag Register	DNFP022EDF3	00_H	8	FFC3 0220_H	3	8
DNF	Edge Detection Control Register	DNFP022EDC4	00_H	8	FFC3 0224_H	3	8
DNF	Edge Detection Flag Register	DNFP022EDF4	00_H	8	FFC3 0228_H	3	8
DNF	Edge Detection Control Register	DNFP022EDC5	00_H	8	FFC3 $022C_H$	3	8
DNF	Edge Detection Flag Register	DNFP022EDF5	00_H	8	FFC3 0230_H	3	8
DNF	Edge Detection Control Register	DNFP022EDC6	00_H	8	FFC3 0234_H	3	8
DNF	Edge Detection Flag Register	DNFP022EDF6	00_H	8	FFC3 0238_H	3	8
DNF	Edge Detection Control Register	DNFP022EDC7	00_H	8	FFC3 $023C_H$	3	8
DNF	Edge Detection Flag Register	DNFP022EDF7	00_H	8	FFC3 0240_H	3	8
DNF	Digital Noise Elimination Control Register	DNFP023CTL	00_H	8	FFC3 0300_H	3	8
DNF	Edge Detection Control Register	DNFP023EDC0	00_H	8	FFC3 0304_H	3	8
DNF	Edge Detection Flag Register	DNFP023EDF0	00_H	8	FFC3 0308_H	3	8
DNF	Edge Detection Control Register	DNFP023EDC1	00_H	8	FFC3 $030C_H$	3	8
DNF	Edge Detection Flag Register	DNFP023EDF1	00_H	8	FFC3 0310_H	3	8
DNF	Edge Detection Control Register	DNFP023EDC2	00_H	8	FFC3 0314_H	3	8
DNF	Edge Detection Flag Register	DNFP023EDF2	00_H	8	FFC3 0318_H	3	8

(22/201)

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
DNF	Edge Detection Control Register	DNFP023EDC3	00_H	8	FFC3 031C$_H$	3	8
DNF	Edge Detection Flag Register	DNFP023EDF3	00_H	8	FFC3 0320$_H$	3	8
DNF	Edge Detection Control Register	DNFP023EDC4	00_H	8	FFC3 0324$_H$	3	8
DNF	Edge Detection Flag Register	DNFP023EDF4	00_H	8	FFC3 0328$_H$	3	8
DNF	Edge Detection Control Register	DNFP023EDC5	00_H	8	FFC3 032C$_H$	3	8
DNF	Edge Detection Flag Register	DNFP023EDF5	00_H	8	FFC3 0330$_H$	3	8
DNF	Edge Detection Control Register	DNFP023EDC6	00_H	8	FFC3 0334$_H$	3	8
DNF	Edge Detection Flag Register	DNFP023EDF6	00_H	8	FFC3 0338$_H$	3	8
DNF	Edge Detection Control Register	DNFP023EDC7	00_H	8	FFC3 033C$_H$	3	8
DNF	Edge Detection Flag Register	DNFP023EDF7	00_H	8	FFC3 0340$_H$	3	8
DNF	Digital Noise Elimination Control Register	DNFP024CTL	00_H	8	FFC3 0400$_H$	3	8
DNF	Edge Detection Control Register	DNFP024EDC0	00_H	8	FFC3 0404$_H$	3	8
DNF	Edge Detection Flag Register	DNFP024EDF0	00_H	8	FFC3 0408$_H$	3	8
DNF	Edge Detection Control Register	DNFP024EDC1	00_H	8	FFC3 040C$_H$	3	8
DNF	Edge Detection Flag Register	DNFP024EDF1	00_H	8	FFC3 0410$_H$	3	8
DNF	Edge Detection Control Register	DNFP024EDC2	00_H	8	FFC3 0414$_H$	3	8
DNF	Edge Detection Flag Register	DNFP024EDF2	00_H	8	FFC3 0418$_H$	3	8
DNF	Edge Detection Control Register	DNFP024EDC3	00_H	8	FFC3 041C$_H$	3	8
DNF	Edge Detection Flag Register	DNFP024EDF3	00_H	8	FFC3 0420$_H$	3	8
DNF	Edge Detection Control Register	DNFP024EDC4	00_H	8	FFC3 0424$_H$	3	8
DNF	Edge Detection Flag Register	DNFP024EDF4	00_H	8	FFC3 0428$_H$	3	8
DNF	Edge Detection Control Register	DNFP024EDC5	00_H	8	FFC3 042C$_H$	3	8
DNF	Edge Detection Flag Register	DNFP024EDF5	00_H	8	FFC3 0430$_H$	3	8
DNF	Edge Detection Control Register	DNFP024EDC6	00_H	8	FFC3 0434$_H$	3	8
DNF	Edge Detection Flag Register	DNFP024EDF6	00_H	8	FFC3 0438$_H$	3	8
DNF	Edge Detection Control Register	DNFP024EDC7	00_H	8	FFC3 043C$_H$	3	8
DNF	Edge Detection Flag Register	DNFP024EDF7	00_H	8	FFC3 0440$_H$	3	8
DNF	Edge Detection Control Register	DNFP024EDC8	00_H	8	FFC3 0444$_H$	3	8
DNF	Edge Detection Flag Register	DNFP024EDF8	00_H	8	FFC3 0448$_H$	3	8
DNF	Edge Detection Control Register	DNFP024EDC9	00_H	8	FFC3 044C$_H$	3	8
DNF	Edge Detection Flag Register	DNFP024EDF9	00_H	8	FFC3 0450$_H$	3	8
DNF	Edge Detection Control Register	DNFP024EDC10	00_H	8	FFC3 0454$_H$	3	8
DNF	Edge Detection Flag Register	DNFP024EDF10	00_H	8	FFC3 0458$_H$	3	8
DNF	Edge Detection Control Register	DNFP024EDC11	00_H	8	FFC3 045C$_H$	3	8
DNF	Edge Detection Flag Register	DNFP024EDF11	00_H	8	FFC3 0460$_H$	3	8
DNF	Edge Detection Control Register	DNFP024EDC12	00_H	8	FFC3 0464$_H$	3	8
DNF	Edge Detection Flag Register	DNFP024EDF12	00_H	8	FFC3 0468$_H$	3	8
DNF	Edge Detection Control Register	DNFP024EDC13	00_H	8	FFC3 046C$_H$	3	8
DNF	Edge Detection Flag Register	DNFP024EDF13	00_H	8	FFC3 0470$_H$	3	8
DNF	Edge Detection Control Register	DNFP024EDC14	00_H	8	FFC3 0474$_H$	3	8
DNF	Edge Detection Flag Register	DNFP024EDF14	00_H	8	FFC3 0478$_H$	3	8
DNF	Edge Detection Control Register	DNFP024EDC15	00_H	8	FFC3 047C$_H$	3	8
DNF	Edge Detection Flag Register	DNFP024EDF15	00_H	8	FFC3 0480$_H$	3	8
DNF	Digital Noise Elimination Control Register	DNFP025CTL	00_H	8	FFC3 0500$_H$	3	8
DNF	Edge Detection Control Register	DNFP025EDC0	00_H	8	FFC3 0504$_H$	3	8
DNF	Edge Detection Flag Register	DNFP025EDF0	00_H	8	FFC3 0508$_H$	3	8

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
DNF	Edge Detection Control Register	DNFP025EDC1	00_H	8	FFC3 $050C_H$	3	8
DNF	Edge Detection Flag Register	DNFP025EDF1	00_H	8	FFC3 0510_H	3	8
DNF	Edge Detection Control Register	DNFP025EDC2	00_H	8	FFC3 0514_H	3	8
DNF	Edge Detection Flag Register	DNFP025EDF2	00_H	8	FFC3 0518_H	3	8
DNF	Edge Detection Control Register	DNFP025EDC3	00_H	8	FFC3 $051C_H$	3	8
DNF	Edge Detection Flag Register	DNFP025EDF3	00_H	8	FFC3 0520_H	3	8
DNF	Edge Detection Control Register	DNFP025EDC4	00_H	8	FFC3 0524_H	3	8
DNF	Edge Detection Flag Register	DNFP025EDF4	00_H	8	FFC3 0528_H	3	8
DNF	Edge Detection Control Register	DNFP025EDC5	00_H	8	FFC3 $052C_H$	3	8
DNF	Edge Detection Flag Register	DNFP025EDF5	00_H	8	FFC3 0530_H	3	8
DNF	Edge Detection Control Register	DNFP025EDC6	00_H	8	FFC3 0534_H	3	8
DNF	Edge Detection Flag Register	DNFP025EDF6	00_H	8	FFC3 0538_H	3	8
DNF	Edge Detection Control Register	DNFP025EDC7	00_H	8	FFC3 $053C_H$	3	8
DNF	Edge Detection Flag Register	DNFP025EDF7	00_H	8	FFC3 0540_H	3	8
DNF	Edge Detection Control Register	DNFP025EDC8	00_H	8	FFC3 0544_H	3	8
DNF	Edge Detection Flag Register	DNFP025EDF8	00_H	8	FFC3 0548_H	3	8
DNF	Edge Detection Control Register	DNFP025EDC9	00_H	8	FFC3 $054C_H$	3	8
DNF	Edge Detection Flag Register	DNFP025EDF9	00_H	8	FFC3 0550_H	3	8
DNF	Edge Detection Control Register	DNFP025EDC10	00_H	8	FFC3 0554_H	3	8
DNF	Edge Detection Flag Register	DNFP025EDF10	00_H	8	FFC3 0558_H	3	8
DNF	Edge Detection Control Register	DNFP025EDC11	00_H	8	FFC3 $055C_H$	3	8
DNF	Edge Detection Flag Register	DNFP025EDF11	00_H	8	FFC3 0560_H	3	8
DNF	Digital Noise Elimination Control Register	DNFP026CTL	00_H	8	FFC3 0600_H	3	8
DNF	Edge Detection Control Register	DNFP026EDC0	00_H	8	FFC3 0604_H	3	8
DNF	Edge Detection Flag Register	DNFP026EDF0	00_H	8	FFC3 0608_H	3	8
DNF	Edge Detection Control Register	DNFP026EDC1	00_H	8	FFC3 $060C_H$	3	8
DNF	Edge Detection Flag Register	DNFP026EDF1	00_H	8	FFC3 0610_H	3	8
DNF	Edge Detection Control Register	DNFP026EDC2	00_H	8	FFC3 0614_H	3	8
DNF	Edge Detection Flag Register	DNFP026EDF2	00_H	8	FFC3 0618_H	3	8
DNF	Edge Detection Control Register	DNFP026EDC3	00_H	8	FFC3 $061C_H$	3	8
DNF	Edge Detection Flag Register	DNFP026EDF3	00_H	8	FFC3 0620_H	3	8
DNF	Edge Detection Control Register	DNFP026EDC4	00_H	8	FFC3 0624_H	3	8
DNF	Edge Detection Flag Register	DNFP026EDF4	00_H	8	FFC3 0628_H	3	8
DNF	Edge Detection Control Register	DNFP026EDC5	00_H	8	FFC3 $062C_H$	3	8
DNF	Edge Detection Flag Register	DNFP026EDF5	00_H	8	FFC3 0630_H	3	8
DNF	Edge Detection Control Register	DNFP026EDC6	00_H	8	FFC3 0634_H	3	8
DNF	Edge Detection Flag Register	DNFP026EDF6	00_H	8	FFC3 0638_H	3	8
DNF	Edge Detection Control Register	DNFP026EDC7	00_H	8	FFC3 $063C_H$	3	8
DNF	Edge Detection Flag Register	DNFP026EDF7	00_H	8	FFC3 0640_H	3	8
DNF	Edge Detection Control Register	DNFP026EDC8	00_H	8	FFC3 0644_H	3	8
DNF	Edge Detection Flag Register	DNFP026EDF8	00_H	8	FFC3 0648_H	3	8
DNF	Edge Detection Control Register	DNFP026EDC9	00_H	8	FFC3 $064C_H$	3	8
DNF	Edge Detection Flag Register	DNFP026EDF9	00_H	8	FFC3 0650_H	3	8
DNF	Edge Detection Control Register	DNFP026EDC10	00_H	8	FFC3 0654_H	3	8
DNF	Edge Detection Flag Register	DNFP026EDF10	00_H	8	FFC3 0658_H	3	8
DNF	Edge Detection Control Register	DNFP026EDC11	00_H	8	FFC3 $065C_H$	3	8

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
DNF	Edge Detection Flag Register	DNFP026EDF11	00_H	8	FFC3 0660_H	3	8
DNF	Edge Detection Control Register	DNFP026EDC12	00_H	8	FFC3 0664_H	3	8
DNF	Edge Detection Flag Register	DNFP026EDF12	00_H	8	FFC3 0668_H	3	8
DNF	Edge Detection Control Register	DNFP026EDC13	00_H	8	FFC3 $066C_H$	3	8
DNF	Edge Detection Flag Register	DNFP026EDF13	00_H	8	FFC3 0670_H	3	8
DNF	Edge Detection Control Register	DNFP026EDC14	00_H	8	FFC3 0674_H	3	8
DNF	Edge Detection Flag Register	DNFP026EDF14	00_H	8	FFC3 0678_H	3	8
DNF	Edge Detection Control Register	DNFP026EDC15	00_H	8	FFC3 $067C_H$	3	8
DNF	Edge Detection Flag Register	DNFP026EDF15	00_H	8	FFC3 0680_H	3	8
DNF	Digital Noise Elimination Control Register	DNFP027CTL	00_H	8	FFC3 0700_H	3	8
DNF	Edge Detection Control Register	DNFP027EDC0	00_H	8	FFC3 0704_H	3	8
DNF	Edge Detection Flag Register	DNFP027EDF0	00_H	8	FFC3 0708_H	3	8
DNF	Edge Detection Control Register	DNFP027EDC1	00_H	8	FFC3 $070C_H$	3	8
DNF	Edge Detection Flag Register	DNFP027EDF1	00_H	8	FFC3 0710_H	3	8
DNF	Edge Detection Control Register	DNFP027EDC2	00_H	8	FFC3 0714_H	3	8
DNF	Edge Detection Flag Register	DNFP027EDF2	00_H	8	FFC3 0718_H	3	8
DNF	Edge Detection Control Register	DNFP027EDC3	00_H	8	FFC3 $071C_H$	3	8
DNF	Edge Detection Flag Register	DNFP027EDF3	00_H	8	FFC3 0720_H	3	8
DNF	Edge Detection Control Register	DNFP027EDC4	00_H	8	FFC3 0724_H	3	8
DNF	Edge Detection Flag Register	DNFP027EDF4	00_H	8	FFC3 0728_H	3	8
DNF	Edge Detection Control Register	DNFP027EDC5	00_H	8	FFC3 $072C_H$	3	8
DNF	Edge Detection Flag Register	DNFP027EDF5	00_H	8	FFC3 0730_H	3	8
DNF	Edge Detection Control Register	DNFP027EDC6	00_H	8	FFC3 0734_H	3	8
DNF	Edge Detection Flag Register	DNFP027EDF6	00_H	8	FFC3 0738_H	3	8
DNF	Edge Detection Control Register	DNFP027EDC7	00_H	8	FFC3 $073C_H$	3	8
DNF	Edge Detection Flag Register	DNFP027EDF7	00_H	8	FFC3 0740_H	3	8
DNF	Edge Detection Control Register	DNFP027EDC8	00_H	8	FFC3 0744_H	3	8
DNF	Edge Detection Flag Register	DNFP027EDF8	00_H	8	FFC3 0748_H	3	8
PBG	PBG3A Protection Register 0	FSGD3ADPROT0	07FF $FFFF_H$	32	FFC4 0000_H	3	8, 16, 32
PBG	PBG3A Protection Register 1	FSGD3ADPROT1	07FF $FFFF_H$	32	FFC4 0004_H	3	8, 16, 32
PBG	PBG3A Protection Register 2	FSGD3ADPROT2	07FF $FFFF_H$	32	FFC4 0008_H	3	8, 16, 32
PBG	PBG3A Protection Register 3	FSGD3ADPROT3	07FF $FFFF_H$	32	FFC4 $000C_H$	3	8, 16, 32
PBG	PBG3A Protection Register 4	FSGD3ADPROT4	07FF $FFFF_H$	32	FFC4 0010_H	3	8, 16, 32
PBG	PBG3A Protection Register 5	FSGD3ADPROT5	07FF $FFFF_H$	32	FFC4 0014_H	3	8, 16, 32
PBG	PBG3A Protection Register 6	FSGD3ADPROT6	07FF $FFFF_H$	32	FFC4 0018_H	3	8, 16, 32
PBG	PBG3A Protection Register 7	FSGD3ADPROT7	07FF $FFFF_H$	32	FFC4 $001C_H$	3	8, 16, 32
PBG	PBG3A Protection Register 8	FSGD3ADPROT8	07FF $FFFF_H$	32	FFC4 0020_H	3	8, 16, 32
PBG	PBG3A Protection Register 9	FSGD3ADPROT9	07FF $FFFF_H$	32	FFC4 0024_H	3	8, 16, 32
PBG	PBG3A Protection Register 10	FSGD3ADPROT10	07FF $FFFF_H$	32	FFC4 0028_H	3	8, 16, 32
PBG	PBG3A Protection Register 11	FSGD3ADPROT11	07FF $FFFF_H$	32	FFC4 $002C_H$	3	8, 16, 32
PBG	PBG3A Protection Register 12	FSGD3ADPROT12	07FF $FFFF_H$	32	FFC4 0030_H	3	8, 16, 32
PBG	PBG3A Protection Register 13	FSGD3ADPROT13	07FF $FFFF_H$	32	FFC4 0034_H	3	8, 16, 32
PBG	PBG3A Protection Register 14	FSGD3ADPROT14	07FF $FFFF_H$	32	FFC4 0038_H	3	8, 16, 32
PBG	PBG3A Protection Register 15	FSGD3ADPROT15	07FF $FFFF_H$	32	FFC4 $003C_H$	3	8, 16, 32
PBG	PBG3A Error Control Register	ERRSLV3ACTL	0000 0000_H	32	FFC4 0200_H	3	8, 16, 32
PBG	PBG3A Error Status Register	ERRSLV3ASTAT	0000 0000_H	32	FFC4 0204_H	3	8, 16, 32

(25/201)

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
PBG	PBG3A Error Address Register	ERRSLV3AADDR	0000 0000$_H$	32	FFC4 0208$_H$	3	32
PBG	PBG3A Error Type Register	ERRSLV3ATYPE	0000 0000$_H$	32	FFC4 020C$_H$	3	16, 32
PBG	PBG3B Protection Register 0	FSGD3BDPROT0	07FF FFFF$_H$	32	FFC4 1000$_H$	3	8, 16, 32
PBG	PBG3B Protection Register 1	FSGD3BDPROT1	07FF FFFF$_H$	32	FFC4 1004$_H$	3	8, 16, 32
PBG	PBG3B Error Control Register	ERRSLV3BCTL	0000 0000$_H$	32	FFC4 1200$_H$	3	8, 16, 32
PBG	PBG3B Error Status Register	ERRSLV3BSTAT	0000 0000$_H$	32	FFC4 1204$_H$	3	8, 16, 32
PBG	PBG3B Error Address Register	ERRSLV3BADDR	0000 0000$_H$	32	FFC4 1208$_H$	3	32
PBG	PBG3B Error Type Register	ERRSLV3BTYPE	0000 0000$_H$	32	FFC4 120C$_H$	3	16, 32
MGDGR	GRG Protection Setting Register 0	MGDGRPROT0	07FF FFF0$_H$	32	FFC4 9000$_H$	0	8, 16, 32
MGDGR	GRG Comparison Base Address Register 0	MGDGRBAD0	0000 0000$_H$	32	FFC4 9004$_H$	0	8, 16, 32
MGDGR	GRG Comparison Address Enable Register 0	MGDGRADV0	0000 0000$_H$	32	FFC4 9008$_H$	0	8, 16, 32
MGDGR	GRG Protection Setting Register 1	MGDGRPROT1	07FF FFF0$_H$	32	FFC4 9010$_H$	0	8, 16, 32
MGDGR	GRG Comparison Base Address Register 1	MGDGRBAD1	0000 0000$_H$	32	FFC4 9014$_H$	0	8, 16, 32
MGDGR	GRG Comparison Address Enable Register 1	MGDGRADV1	0000 0000$_H$	32	FFC4 9018$_H$	0	8, 16, 32
MGDGR	GRG Protection Setting Register 2	MGDGRPROT2	07FF FFF0$_H$	32	FFC4 9020$_H$	0	8, 16, 32
MGDGR	GRG Comparison Base Address Register 2	MGDGRBAD2	0000 0000$_H$	32	FFC4 9024$_H$	0	8, 16, 32
MGDGR	GRG Comparison Address Enable Register 2	MGDGRADV2	0000 0000$_H$	32	FFC4 9028$_H$	0	8, 16, 32
MGDGR	GRG Protection Setting Register 3	MGDGRPROT3	07FF FFF0$_H$	32	FFC4 9030$_H$	0	8, 16, 32
MGDGR	GRG Comparison Base Address Register 3	MGDGRBAD3	0000 0000$_H$	32	FFC4 9034$_H$	0	8, 16, 32
MGDGR	GRG Comparison Address Enable Register 3	MGDGRADV3	0000 0000$_H$	32	FFC4 9038$_H$	0	8, 16, 32
MGDGR	GRG Control Register	MGDGRSCTL_VCI	0000 0000$_H$	32	FFC4 9100$_H$	0	8, 16, 32
MGDGR	GRG Error Status Register	MGDGRSSTAT_VCI	0000 0000$_H$	32	FFC4 9104$_H$	0	8, 16, 32
MGDGR	GRG Error Address Register	MGDGRSAD_VCI	0000 0000$_H$	32	FFC4 9108$_H$	0	8, 16, 32
MGDGR	GRG Error Access Type Register	MGDGRSTYPE_VCI	0000 0000$_H$	32	FFC4 910C$_H$	0	8, 16, 32
MGDGR	GRG Control Register	MGDGRSCTL_PE1	0000 0000$_H$	32	FFC4 9200$_H$	0	8, 16, 32
MGDGR	GRG Error Status Register	MGDGRSSTAT_PE1	0000 0000$_H$	32	FFC4 9204$_H$	0	8, 16, 32
MGDGR	GRG Error Address Register	MGDGRSAD_PE1	0000 0000$_H$	32	FFC4 9208$_H$	0	8, 16, 32
MGDGR	GRG Error Access Type Register	MGDGRSTYPE_PE1	0000 0000$_H$	32	FFC4 920C$_H$	0	8, 16, 32
PBG	PBG0A Protection Register 0	FSGD0ADPROT0	07FF FFFF$_H$	32	FFC4 C000$_H$	0	8, 16, 32
PBG	PBG0A Protection Register 1	FSGD0ADPROT1	07FF FFFF$_H$	32	FFC4 C004$_H$	0	8, 16, 32
PBG	PBG0B Protection Register 4	FSGD0BDPROT4	07FF FFFF$_H$	32	FFC4 C110$_H$	0	8, 16, 32
PBG	PBG0B Protection Register 5	FSGD0BDPROT5	07FF FFFF$_H$	32	FFC4 C114$_H$	0	8, 16, 32
PBG	PBG0B Protection Register 6	FSGD0BDPROT6	07FF FFFF$_H$	32	FFC4 C118$_H$	0	8, 16, 32
PBG	PBG0A Error Control Register	ERRSLV0ACTL	0000 0000$_H$	32	FFC4 C800$_H$	0	8, 16, 32
PBG	PBG0A Error Status Register	ERRSLV0ASTAT	0000 0000$_H$	32	FFC4 C804$_H$	0	8, 16, 32
PBG	PBG0A Error Address Register	ERRSLV0AADDR	0000 0000$_H$	32	FFC4 C808$_H$	0	32
PBG	PBG0A Error Type Register	ERRSLV0ATYPE	0000 0000$_H$	32	FFC4 C80C$_H$	0	16, 32
PBG	PBG0B Error Control Register	ERRSLV0BCTL	0000 0000$_H$	32	FFC4 C900$_H$	0	8, 16, 32
PBG	PBG0B Error Status Register	ERRSLV0BSTAT	0000 0000$_H$	32	FFC4 C904$_H$	0	8, 16, 32
PBG	PBG0B Error Address Register	ERRSLV0BADDR	0000 0000$_H$	32	FFC4 C908$_H$	0	32
PBG	PBG0B Error Type Register	ERRSLV0BTYPE	0000 0000$_H$	32	FFC4 C90C$_H$	0	16, 32
MISG	MISR Comparator Control Register	MISRCMPCTL	0000$_H$	16	FFC5 0000$_H$	0	8, 16

(26/201)

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
MISG	MISR Comparison Error Status Register	MISRCMPERSTR	00_H	8	FFC5 0004_H	0	8
MISG	MISR Comparison Error Status Clear Register	MISRCMPERRSTC	00_H	8	FFC5 0008_H	0	8
MISG	MISR Error Notification Control Register	MISRERRCTL	00_H	8	FFC5 $000C_H$	0	8
MISG	Multi Input Signature Register 1L (PE1)	MISR1L_PE1	0000 0000_H	32	FFC5 1000_H	0	32
MISG	Multi Input Signature Register 1H (PE1)	MISR1H_PE1	0000 0000_H	32	FFC5 1004_H	0	32
MISG	Multi Input Signature Register 2L (PE1)	MISR2L_PE1	0000 0000_H	32	FFC5 1008_H	0	32
MISG	Multi Input Signature Register 2H (PE1)	MISR2H_PE1	0000 0000_H	32	FFC5 $100C_H$	0	32
MISG	MISR Calculation Data Register (PE1)	MISRCDRL_PE1	—	32	FFC5 1010_H	0	8, 16, 32
MISG	MISR Control Register (PE1)	MISRCR_PE1	00_H	8	FFC5 1018_H	0	8
MISG	MISR Monitoring Area Base Address Register (PE1)	MISRBASEADR_PE1	0000 0000_H	32	FFC5 $101C_H$	0	8, 16, 32
MISG	MISR Monitoring Area Address Mask Register (PE1)	MISRADRMSK_PE1	0000 0000_H	32	FFC5 1020_H	0	8, 16, 32
MISG	MISR Data Counter Control Register (PE1)	MISRDCNTCTL_PE1	00_H	8	FFC5 1024_H	0	8
MISG	MISR Data Counter Register (PE1)	MISRDCNT_PE1	0000_H	16	FFC5 1028_H	0	16
MISG	Multi Input Signature Register 1L (PCU)	MISR1L_PCU	0000 0000_H	32	FFC5 3000_H	0	32
MISG	Multi Input Signature Register 1H (PCU)	MISR1H_PCU	0000 0000_H	32	FFC5 3004_H	0	32
MISG	Multi Input Signature Register 2L (PCU)	MISR2L_PCU	0000 0000_H	32	FFC5 3008_H	0	32
MISG	Multi Input Signature Register 2H (PCU)	MISR2H_PCU	0000 0000_H	32	FFC5 $300C_H$	0	32
MISG	MISR Calculation Data Register (PCU)	MISRCDRL_PCU	—	32	FFC5 3010_H	0	8, 16, 32
MISG	MISR Control Register (PCU)	MISRCR_PCU	00_H	8	FFC5 3018_H	0	8
MISG	MISR Monitoring Area Base Address Register (PCU)	MISRBASEADR_PCU	0000 0000_H	32	FFC5 $301C_H$	0	8, 16, 32
MISG	MISR Monitoring Area Address Mask Register (PCU)	MISRADRMSK_PCU	0000 0000_H	32	FFC5 3020_H	0	8, 16, 32
MISG	MISR Data Counter Control Register (PCU)	MISRDCNTCTL_PCU	00_H	8	FFC5 3024_H	0	8
MISG	MISR Data Counter Register (PCU)	MISRDCNT_PCU	0000_H	16	FFC5 3028_H	0	16
FLASH	Data Flash Memory Read Cycle Setting Register	FRDCYCLD	$0F_H$	8	FFC5 9810_H	0	8
ECCIC1	Instruction Cache Data RAM ECC Control Register	IDECCCTL_PE1	0000 0000_H	32	FFC6 0400_H	0	16, 32
ECCIC1	Instruction Cache Data RAM Error Information Control Register	IDERRINT_PE1	0000 0000_H	32	FFC6 0404_H	0	8, 16, 32
ECCIC1	Instruction Cache Data RAM Error Status Clear Register	IDSTCLR_PE1	0000 0000_H	32	FFC6 0408_H	0	8, 16, 32
ECCIC1	Instruction Cache Data RAM Error Count Overflow Status Register	IDOVFSTR_PE1	0000 0000_H	32	FFC6 $040C_H$	0	8, 16, 32
ECCIC1	Instruction Cache Data RAM 1st Error Status Register	ID1STERSTR_PE1	0000 0000_H	32	FFC6 0410_H	0	8, 16, 32
ECCIC1	Instruction Cache Data RAM (Bank0) 1st Error Address Register	ID1STEADR0_PE1	0000 0000_H	32	FFC6 0450_H	0	8, 16, 32
ECCIC1	Instruction Cache Data RAM (Bank1) 1st Error Address Register	ID1STEADR1_PE1	0000 0000_H	32	FFC6 0454_H	0	8, 16, 32
ECCIC1	Instruction Cache Tag RAM ECC Control Register	ITECCCTL_PE1	0000 0000_H	32	FFC6 1400_H	0	16, 32
ECCIC1	Instruction Cache Tag RAM Error Information Control Register	ITERRINT_PE1	0000 0000_H	32	FFC6 1404_H	0	8, 16, 32
ECCIC1	Instruction Cache Tag RAM Error Status Clear Register	ITSTCLR_PE1	0000 0000_H	32	FFC6 1408_H	0	8, 16, 32
ECCIC1	Instruction Cache Tag RAM Error Count Overflow Status Register	ITOVFSTR_PE1	0000 0000_H	32	FFC6 $140C_H$	0	8, 16, 32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
ECCIC1	Instruction Cache Tag RAM 1st Error Status Register	IT1STERSTR_PE1	0000 0000$_H$	32	FFC6 1410$_H$	0	8, 16, 32
ECCIC1	Instruction Cache Tag RAM (Bank0) 1st Error Address Register	IT1STEADR0_PE1	0000 0000$_H$	32	FFC6 1450$_H$	0	8, 16, 32
ECCFLI	Code Flash Address Parity Control Register	CFAPCTL	0000 0000$_H$	32	FFC6 2000$_H$	0	16, 32
ECCFLI	Code Flash ECC Control Register	CFECCCTL_VCI	0000 0000$_H$	32	FFC6 2200$_H$	0	16, 32
ECCFLI	Code Flash Error Information Control Register	CFERRINT_VCI	0000 0006$_H$	32	FFC6 2204$_H$	0	8, 16, 32
ECCFLI	Code Flash Status Clear Register	CFSTCLR_VCI	0000 0000$_H$	32	FFC6 2208$_H$	0	8, 16, 32
ECCFLI	Code Flash Error Count Overflow Status Register	CFOVFSTR_VCI	0000 0000$_H$	32	FFC6 220C$_H$	0	8, 16, 32
ECCFLI	Code Flash 1st Error Status Register	CF1STERSTR_VCI	0000 0000$_H$	32	FFC6 2210$_H$	0	8, 16, 32
ECCFLI	Code Flash 1st Error Address Register	CF1STEADR0_VCI	0000 0000$_H$	32	FFC6 2250$_H$	0	8, 16, 32
ECCFLI	Code Flash Sub Test Control Register (VCI)	CFSTSTCTL_VCI	0000 0000$_H$	32	FFC6 2350$_H$	0	16, 32
ECCFLI	Code Flash ECC Control Register	CFECCCTL_PE1	0000 0000$_H$	32	FFC6 2400$_H$	0	16, 32
ECCFLI	Code Flash Error Information Control Register	CFERRINT_PE1	0000 0006$_H$	32	FFC6 2404$_H$	0	8, 16, 32
ECCFLI	Code Flash Status Clear Register	CFSTCLR_PE1	0000 0000$_H$	32	FFC6 2408$_H$	0	8, 16, 32
ECCFLI	Code Flash Error Count Overflow Status Register	CFOVFSTR_PE1	0000 0000$_H$	32	FFC6 240C$_H$	0	8, 16, 32
ECCFLI	Code Flash 1st Error Status Register	CF1STERSTR_PE1	0000 0000$_H$	32	FFC6 2410$_H$	0	8, 16, 32
ECCFLI	Code Flash 1st Error Address Register	CF1STEADR0_PE1	0000 0000$_H$	32	FFC6 2450$_H$	0	8, 16, 32
ECCFLI	Code Flash Sub Test Control Register (PE1)	CFSTSTCTL_PE1	0000 0000$_H$	32	FFC6 2550$_H$	0	16, 32
ECCFLI	Code Flash ECC Control Register	CFECCCTL_PCU	0000 0000$_H$	32	FFC6 2800$_H$	0	16, 32
ECCFLI	Code Flash Error Information Control Register	CFERRINT_PCU	0000 0006$_H$	32	FFC6 2804$_H$	0	8, 16, 32
ECCFLI	Code Flash Status Clear Register	CFSTCLR_PCU	0000 0000$_H$	32	FFC6 2808$_H$	0	8, 16, 32
ECCFLI	Code Flash Error Count Overflow Status Register	CFOVFSTR_PCU	0000 0000$_H$	32	FFC6 280C$_H$	0	8, 16, 32
ECCFLI	Code Flash 1st Error Status Register	CF1STERSTR_PCU	0000 0000$_H$	32	FFC6 2810$_H$	0	8, 16, 32
ECCFLI	Code Flash 1st Error Address Register	CF1STEADR0_PCU	0000 0000$_H$	32	FFC6 2850$_H$	0	8, 16, 32
ECCFLI	Code Flash Sub Test Control Register (PCU)	CFSTSTCTL_PCU	0000 0000$_H$	32	FFC6 2950$_H$	0	16, 32
ECCEEP	Data Flash ECC Control Register	DFECCCTL	0000$_H$	16	FFC6 2C00$_H$	0	16
ECCEEP	Data Flash Error Status Register	DFERSTR	00$_H$	8	FFC6 2C04$_H$	0	8
ECCEEP	Data Flash Error Status Clear Register	DFERSTC	00$_H$	8	FFC6 2C08$_H$	0	8
ECCEEP	Data Flash Error Overflow Status Register	DFOVFSTR	00$_H$	8	FFC6 2C0C$_H$	0	8
ECCEEP	Data Flash Error Overflow Status Clear Register	DFOVFSTC	00$_H$	8	FFC6 2C10$_H$	0	8
ECCEEP	Data Flash Error Notification Control Register	DFERRINT	02$_H$	8	FFC6 2C14$_H$	0	8
ECCEEP	Data Flash 1st Error Address Register	DFEADR	0000 0000$_H$	32	FFC6 2C18$_H$	0	32
ECCEEP	Data Flash Test Control Register	DFTSTCTL	0000$_H$	16	FFC6 2C1C$_H$	0	16
ECCEEPC	Data Flash ECC Control Register	DFECCCTL	0000$_H$	16	FFC6 2E00$_H$	0	16
ECCEEPC	Data Flash Error Status Register	DFERSTR	00$_H$	8	FFC6 2E04$_H$	0	8
ECCEEPC	Data Flash Error Status Clear Register	DFERSTC	00$_H$	8	FFC6 2E08$_H$	0	8
ECCEEPC	Data Flash Error Overflow Status Register	DFOVFSTR	00$_H$	8	FFC6 2E0C$_H$	0	8
ECCEEPC	Data Flash Error Overflow Status Clear Register	DFOVFSTC	00$_H$	8	FFC6 2E10$_H$	0	8

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
ECCEEPC	Data Flash Error Notification Control Register	DFERRINT	02_H	8	FFC6 $2E14_H$	0	8
ECCEEPC	Data Flash 1st Error Address Register	DFEADR	0000 0000_H	32	FFC6 $2E18_H$	0	32
ECCEEPC	Data Flash Test Control Register	DFTSTCTL	0000_H	16	FFC6 $2E1C_H$	0	16
ECCGRAM	Global RAM ECC Control Register	GRECCCTL_GRAMC	0000 0000_H	32	FFC6 4000_H	0	16, 32
ECCGRAM	Global RAM Test Control Register	GRTSTCTL	0000 0000_H	32	FFC6 4004_H	0	16, 32
ECCGRAM	Global RAM Test Data Read Buffer 0	GRTDATBF0	0000 0000_H	32	FFC6 4008_H	0	8, 16, 32
ECCGRAM	Global RAM Test Data Read Buffer 1	GRTDATBF1	0000 0000_H	32	FFC6 $400C_H$	0	8, 16, 32
ECCGRAM	Global RAM Test Data Read Buffer 2	GRTDATBF2	0000 0000_H	32	FFC6 4010_H	0	8, 16, 32
ECCGRAM	Global RAM Test Data Read Buffer 3	GRTDATBF3	0000 0000_H	32	FFC6 4014_H	0	8, 16, 32
ECCGRAM	Global RAM ECC Decode Input Data Buffer 0	GRDECINBF0	0000 0000_H	32	FFC6 4018_H	0	8, 16, 32
ECCGRAM	Global RAM ECC Decode Input Data Buffer 1	GRDECINBF1	0000 0000_H	32	FFC6 $401C_H$	0	8, 16, 32
ECCGRAM	Global RAM ECC Control Register	GRECCCTL_VCI	0000 0000_H	32	FFC6 4200_H	0	16, 32
ECCGRAM	Global RAM Error Information Control Register	GRERRINT_VCI	0000 0006_H	32	FFC6 4204_H	0	8, 16, 32
ECCGRAM	Global RAM Status Clear Register	GRSTCLR_VCI	0000 0000_H	32	FFC6 4208_H	0	8, 16, 32
ECCGRAM	Global RAM Error Count Overflow Status Register	GROVFSTR_VCI	0000 0000_H	32	FFC6 $420C_H$	0	8, 16, 32
ECCGRAM	Global RAM 1st Error Status Register	GR1STERSTR_VCI	0000 0000_H	32	FFC6 4210_H	0	8, 16, 32
ECCGRAM	Global RAM 1st Error (lower 32 bit Data) Address Register	GR1STEADR0_VCI	0000 0000_H	32	FFC6 4250_H	0	8, 16, 32
ECCGRAM	Global RAM 1st Error (Upper 32-bit Data) Address Register	GR1STEADR1_VCI	0000 0000_H	32	FFC6 4254_H	0	8, 16, 32
ECCGRAM	Global RAM ECC Control Register	GRECCCTL_PE1	0000 0000_H	32	FFC6 4400_H	0	16, 32
ECCGRAM	Global RAM Error Information Control Register	GRERRINT_PE1	0000 0006_H	32	FFC6 4404_H	0	8, 16, 32
ECCGRAM	Global RAM Status Clear Register	GRSTCLR_PE1	0000 0000_H	32	FFC6 4408_H	0	8, 16, 32
ECCGRAM	Global RAM Error Count Overflow Status Register	GROVFSTR_PE1	0000 0000_H	32	FFC6 $440C_H$	0	8, 16, 32
ECCGRAM	Global RAM 1st Error Status Register	GR1STERSTR_PE1	0000 0000_H	32	FFC6 4410_H	0	8, 16, 32
ECCGRAM	Global RAM 1st Error (lower 32-bit Data) Address Register	GR1STEADR0_PE1	0000 0000_H	32	FFC6 4450_H	0	8, 16, 32
ECCGRAM	Global RAM 1st Error (upper 32-bit Data) Address Register	GR1STEADR1_PE1	0000 0000_H	32	FFC6 4454_H	0	8, 16, 32
ECCCPU1	Local RAM Address Parity Control Register	LRAPCTL_PE1	0000 0000_H	32	FFC6 5000_H	0	16, 32
ECCCPU1	Local RAM Test Control Register	LRTSTCTL_PE1	0000 0000_H	32	FFC6 5004_H	0	16, 32
ECCCPU1	Local RAM Test Data Read Buffer 0	LRTDATBF0_PE1	0000 0000_H	32	FFC6 5008_H	0	8, 16, 32
ECCCPU1	Local RAM Test Data Read Buffer 1	LRTDATBF1_PE1	0000 0000_H	32	FFC6 $500C_H$	0	8, 16, 32
ECCCPU1	Local RAM ECC Control Register	LRECCCTL_PE1	0000 0000_H	32	FFC6 5400_H	0	16, 32
ECCCPU1	Local RAM Error Information Control Register	LRERRINT_PE1	0000 0006_H	32	FFC6 5404_H	0	8, 16, 32
ECCCPU1	Local RAM Status Clear Register	LRSTCLR_PE1	0000 0000_H	32	FFC6 5408_H	0	8, 16, 32
ECCCPU1	Local RAM Error Count Overflow Status Register	LROVFSTR_PE1	0000 0000_H	32	FFC6 $540C_H$	0	8, 16, 32
ECCCPU1	Local RAM 1st Error Status Register	LR1STERSTR_PE1	0000 0000_H	32	FFC6 5410_H	0	8, 16, 32
ECCPCU	Local RAM ECC Control Register	LRECCCTL_PCU	0000 0000_H	32	FFC6 5800_H	0	16, 32
ECCPCU	Local RAM 1st Error Status Register	LRFSTERSTR_PCU	0000 0000_H	32	FFC6 5804_H	0	32
ECCPCU	Local RAM Error Status Clear Register	LRSTCLR_PCU	0000 0000_H	32	FFC6 5824_H	0	8, 16, 32
ECCPCU	Local RAM Overflow Status Register	LROVFSTR_PCU	0000 0000_H	32	FFC6 5828_H	0	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
ECCPCU	Local RAM Overflow Status Clear Register	LROVFSTC_PCU	0000 0000$_H$	32	FFC6 582C$_H$	0	8, 16, 32
ECCPCU	Local RAM 1st Error Address Register 0	LRFSTEADR0_PCU	0000 0000$_H$	32	FFC6 5830$_H$	0	32
ECCPCU	Local RAM Error Information Control Register	LRERRINT_PCU	0000 0002$_H$	32	FFC6 58B0$_H$	0	8, 16, 32
ECCPCU	Local RAM Test Control Register (PCU)	LRTSTCTL_PCU	0000 0000$_H$	32	FFC6 58B4$_H$	0	16, 32
ECCPCU	Local RAM Test Data Read Buffer (PCU)	LRTDATBF0_PCU	0000 0000$_H$	32	FFC6 58B8$_H$	0	32
APDP	P-Bus Data Parity Status Register INTC2	APDPERRST_INTC2	0000 0000$_H$	32	FFC6 8800$_H$	0	8, 16, 32
APDP	P-Bus Data Parity Status Clear Register INTC2	APDPERRSTC_INTC2	0000 0000$_H$	32	FFC6 8804$_H$	0	8, 16, 32
APDP	P-Bus Data Parity Test Mode Control Register INTC2	APDPTMC_INTC2	0000 0000$_H$	32	FFC6 8808$_H$	0	16, 32
APDP	P-Bus Data Parity Error Address Register INTC2	APDPERRADR_INTC2	0000 0000$_H$	32	FFC6 880C$_H$	0	32
APDP	P-Bus Data Parity Status Register PDMA	APDPERRST_PDMA	0000 0000$_H$	32	FFC6 8900$_H$	0	8, 16, 32
APDP	P-Bus Data Parity Status Clear Register PDMA	APDPERRSTC_PDMA	0000 0000$_H$	32	FFC6 8904$_H$	0	8, 16, 32
APDP	P-Bus Data Parity Test Mode Control Register PDMA	APDPTMC_PDMA	0000 0000$_H$	32	FFC6 8908$_H$	0	16, 32
APDP	P-Bus Data Parity Error Address Register PDMA	APDPERRADR_PDMA	0000 0000$_H$	32	FFC6 890C$_H$	0	32
E7RC0M	ECC Control Register	E710CTL	001X$_H$	16	FFC7 1000$_H$	2	8, 16
E7RC0M	ECC Test Mode Control Register	E710TMC	0000$_H$	16	FFC7 1004$_H$	2	8, 16
E7RC0M	ECC Redundant Bit Data Control Test Register	E710TRC	0000 0000$_H$	32	FFC7 1008$_H$	2	32
E7RC0M	ECC Redundant Bit Input and Output Substitution Buffer Register	ECERDB	00$_H$	8	FFC7 1008$_H$	2	8
E7RC0M	ECC Encode Test Register	ECECRD	00$_H$	8	FFC7 1009$_H$	2	8
E7RC0M	ECC7 Bit Redundant Bit Data Retention Test Register	ECHORD	00$_H$	8	FFC7 100A$_H$	2	8
E7RC0M	ECC Decode Syndrome Data Register	ECSYND	00$_H$	8	FFC7 100B$_H$	2	8
E7RC0M	ECC Encode / Decode Data Test Register	E710TED	0000 0000$_H$	32	FFC7 100C$_H$	2	32
E7RC0M	ECC Error Address Register	E710EAD	0000 0000$_H$	32	FFC7 1010$_H$	2	32
E7RC0C	ECC Control Register	E710CTL	001X$_H$	16	FFC7 1200$_H$	2	8, 16
E7RC0C	ECC Test Mode Control Register	E710TMC	0000$_H$	16	FFC7 1204$_H$	2	8, 16
E7RC0C	ECC Redundant Bit Data Control Test Register	E710TRC	0000 0000$_H$	32	FFC7 1208$_H$	2	32
E7RC0C	ECC Redundant Bit input and Output Substitution Buffer Register	ECERDB	00$_H$	8	FFC7 1208$_H$	2	8
E7RC0C	ECC Encode Test Register	ECECRD	00$_H$	8	FFC7 1209$_H$	2	8
E7RC0C	ECC7 Bit Redundant Bit Data Retention Test Register	ECHORD	00$_H$	8	FFC7 120A$_H$	2	8
E7RC0C	ECC Decode Syndrome Data Register	ECSYND	00$_H$	8	FFC7 120B$_H$	2	8
E7RC0C	ECC Encode / Decode Data Test Register	E710TED	0000 0000$_H$	32	FFC7 120C$_H$	2	32
E7RC0C	ECC Error Address Register	E710EAD	0000 0000$_H$	32	FFC7 1210$_H$	2	32
E7FR0M	ECC Control Register	E710CTL	001X$_H$	16	FFC7 2000$_H$	2	8, 16
E7FR0M	ECC Test Mode Control Register	E710TMC	0000$_H$	16	FFC7 2004$_H$	2	8, 16
E7FR0M	ECC Redundant Bit Data Control Test Register	E710TRC	0000 0000$_H$	32	FFC7 2008$_H$	2	32
E7FR0M	ECC Redundant Bit input and Output Substitution Buffer Register	ECERDB	00$_H$	8	FFC7 2008$_H$	2	8

(30/201)

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
E7FR0M	ECC Encode Test Register	ECECRD	00_H	8	FFC7 2009_H	2	8
E7FR0M	ECC7 Bit Redundant Bit Data Retention Test Register	ECHORD	00_H	8	FFC7 $200A_H$	2	8
E7FR0M	ECC Decode Syndrome Data Register	ECSYND	00_H	8	FFC7 $200B_H$	2	8
E7FR0M	ECC Encode / Decode Data Test Register	E710TED	0000 0000_H	32	FFC7 $200C_H$	2	32
E7FR0M	ECC Error Address Register	E710EAD	0000 0000_H	32	FFC7 2010_H	2	32
E7FR0C	ECC Control Register	E710CTL	$001X_H$	16	FFC7 2200_H	2	8, 16
E7FR0C	ECC Test Mode Control Register	E710TMC	0000_H	16	FFC7 2204_H	2	8, 16
E7FR0C	ECC Redundant Bit Data Control Test Register	E710TRC	0000 0000_H	32	FFC7 2208_H	2	32
E7FR0C	ECC Redundant Bit input and Output Substitution Buffer Register	ECERDB	00_H	8	FFC7 2208_H	2	8
E7FR0C	ECC Encode Test Register	ECECRD	00_H	8	FFC7 2209_H	2	8
E7FR0C	ECC7 Bit Redundant Bit Data Retention Test Register	ECHORD	00_H	8	FFC7 $220A_H$	2	8
E7FR0C	ECC Decode Syndrome Data Register	ECSYND	00_H	8	FFC7 $220B_H$	2	8
E7FR0C	ECC Encode / Decode Data Test Register	E710TED	0000 0000_H	32	FFC7 $220C_H$	2	32
E7FR0C	ECC Error Address Register	E710EAD	0000 0000_H	32	FFC7 2210_H	2	32
E7FR1M	ECC Control Register	E710CTL	$001X_H$	16	FFC7 2400_H	2	8, 16
E7FR1M	ECC Test Mode Control Register	E710TMC	0000_H	16	FFC7 2404_H	2	8, 16
E7FR1M	ECC Redundant Bit Data Control Test Register	E710TRC	0000 0000_H	32	FFC7 2408_H	2	32
E7FR1M	ECC Redundant Bit input and Output Substitution Buffer Register	ECERDB	00_H	8	FFC7 2408_H	2	8
E7FR1M	ECC Encode Test Register	ECECRD	00_H	8	FFC7 2409_H	2	8
E7FR1M	ECC7 Bit Redundant Bit Data Retention Test Register	ECHORD	00_H	8	FFC7 $240A_H$	2	8
E7FR1M	ECC Decode Syndrome Data Register	ECSYND	00_H	8	FFC7 $240B_H$	2	8
E7FR1M	ECC Encode / Decode Data Test Register	E710TED	0000 0000_H	32	FFC7 $240C_H$	2	32
E7FR1M	ECC Error Address Register	E710EAD	0000 0000_H	32	FFC7 2410_H	2	32
E7FR1C	ECC Control Register	E710CTL	$001X_H$	16	FFC7 2600_H	2	8, 16
E7FR1C	ECC Test Mode Control Register	E710TMC	0000_H	16	FFC7 2604_H	2	8, 16
E7FR1C	ECC Redundant Bit Data Control Test Register	E710TRC	0000 0000_H	32	FFC7 2608_H	2	32
E7FR1C	ECC Redundant Bit input and Output Substitution Buffer Register	ECERDB	00_H	8	FFC7 2608_H	2	8
E7FR1C	ECC Encode Test Register	ECECRD	00_H	8	FFC7 2609_H	2	8
E7FR1C	ECC7 Bit Redundant Bit Data Retention Test Register	ECHORD	00_H	8	FFC7 $260A_H$	2	8
E7FR1C	ECC Decode Syndrome Data Register	ECSYND	00_H	8	FFC7 $260B_H$	2	8
E7FR1C	ECC Encode / Decode Data Test Register	E710TED	0000 0000_H	32	FFC7 $260C_H$	2	32
E7FR1C	ECC Error Address Register	E710EAD	0000 0000_H	32	FFC7 2610_H	2	32
E7FR2M	ECC Control Register	E710CTL	$001X_H$	16	FFC7 2800_H	2	8, 16
E7FR2M	ECC Test Mode Control Register	E710TMC	0000_H	16	FFC7 2804_H	2	8, 16
E7FR2M	ECC Redundant Bit Data Control Test Register	E710TRC	0000 0000_H	32	FFC7 2808_H	2	32
E7FR2M	ECC Redundant Bit input and Output Substitution Buffer Register	ECERDB	00_H	8	FFC7 2808_H	2	8
E7FR2M	ECC Encode Test Register	ECECRD	00_H	8	FFC7 2809_H	2	8

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
E7FR2M	ECC7 Bit Redundant Bit Data Retention Test Register	ECHORD	00_H	8	FFC7 280A$_H$	2	8
E7FR2M	ECC Decode Syndrome Data Register	ECSYND	00_H	8	FFC7 280B$_H$	2	8
E7FR2M	ECC Encode / Decode Data Test Register	E710TED	0000 0000_H	32	FFC7 280C$_H$	2	32
E7FR2M	ECC Error Address Register	E710EAD	0000 0000_H	32	FFC7 2810_H	2	32
E7FR2C	ECC Control Register	E710CTL	$001X_H$	16	FFC7 $2A00_H$	2	8, 16
E7FR2C	ECC Test Mode Control Register	E710TMC	0000_H	16	FFC7 2A04$_H$	2	8, 16
E7FR2C	ECC Redundant Bit Data Control Test Register	E710TRC	0000 0000_H	32	FFC7 2A08$_H$	2	32
E7FR2C	ECC Redundant Bit input and Output Substitution Buffer Register	ECERDB	00_H	8	FFC7 2A08$_H$	2	8
E7FR2C	ECC Encode Test Register	ECECRD	00_H	8	FFC7 2A09$_H$	2	8
E7FR2C	ECC7 Bit Redundant Bit Data Retention Test Register	ECHORD	00_H	8	FFC7 2A0A$_H$	2	8
E7FR2C	ECC Decode Syndrome Data Register	ECSYND	00_H	8	FFC7 2A0B$_H$	2	8
E7FR2C	ECC Encode / Decode Data Test Register	E710TED	0000 0000_H	32	FFC7 2A0C$_H$	2	32
E7FR2C	ECC Error Address Register	E710EAD	0000 0000_H	32	FFC7 $2A10_H$	2	32
E7AP0M	ECC Control Register	E710CTL	$001X_H$	16	FFC7 6000_H	1(1L)	8, 16
E7AP0M	ECC Test Mode Control Register	E710TMC	0000_H	16	FFC7 6004$_H$	1(1L)	8, 16
E7AP0M	ECC Redundant Bit Data Control Test Register	E710TRC	0000 0000_H	32	FFC7 6008$_H$	1(1L)	32
E7AP0M	ECC Redundant Bit input and Output Substitution Buffer Register	ECERDB	00_H	8	FFC7 6008$_H$	1(1L)	8
E7AP0M	ECC Encode Test Register	ECECRD	00_H	8	FFC7 6009$_H$	1(1L)	8
E7AP0M	ECC7 Bit Redundant Bit Data Retention Test Register	ECHORD	00_H	8	FFC7 600A$_H$	1(1L)	8
E7AP0M	ECC Decode Syndrome Data Register	ECSYND	00_H	8	FFC7 600B$_H$	1(1L)	8
E7AP0M	ECC Encode / Decode Data Test Register	E710TED	0000 0000_H	32	FFC7 600C$_H$	1(1L)	32
E7AP0M	ECC Error Address Register	E710EAD	0000 0000_H	32	FFC7 6010_H	1(1L)	32
E7AP0C	ECC Control Register	E710CTL	$001X_H$	16	FFC7 6200_H	1(1L)	8, 16
E7AP0C	ECC Test Mode Control Register	E710TMC	0000_H	16	FFC7 6204$_H$	1(1L)	8, 16
E7AP0C	ECC Redundant Bit Data Control Test Register	E710TRC	0000 0000_H	32	FFC7 6208$_H$	1(1L)	32
E7AP0C	ECC Redundant Bit input and Output Substitution Buffer Register	ECERDB	00_H	8	FFC7 6208$_H$	1(1L)	8
E7AP0C	ECC Encode Test Register	ECECRD	00_H	8	FFC7 6209$_H$	1(1L)	8
E7AP0C	ECC7 Bit Redundant Bit Data Retention Test Register	ECHORD	00_H	8	FFC7 620A$_H$	1(1L)	8
E7AP0C	ECC Decode Syndrome Data Register	ECSYND	00_H	8	FFC7 620B$_H$	1(1L)	8
E7AP0C	ECC Encode / Decode Data Test Register	E710TED	0000 0000_H	32	FFC7 620C$_H$	1(1L)	32
E7AP0C	ECC Error Address Register	E710EAD	0000 0000_H	32	FFC7 6210_H	1(1L)	32
E7AP1M	ECC Control Register	E710CTL	$001X_H$	16	FFC7 6400_H	1(1L)	8, 16
E7AP1M	ECC Test Mode Control Register	E710TMC	0000_H	16	FFC7 6404$_H$	1(1L)	8, 16
E7AP1M	ECC Redundant Bit Data Control Test Register	E710TRC	0000 0000_H	32	FFC7 6408$_H$	1(1L)	32
E7AP1M	ECC Redundant Bit input and Output Substitution Buffer Register	ECERDB	00_H	8	FFC7 6408$_H$	1(1L)	8
E7AP1M	ECC Encode Test Register	ECECRD	00_H	8	FFC7 6409$_H$	1(1L)	8
E7AP1M	ECC7 Bit Redundant Bit Data Retention Test Register	ECHORD	00_H	8	FFC7 640A$_H$	1(1L)	8

(32/201)

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
E7AP1M	ECC Decode Syndrome Data Register	ECSYND	00_H	8	FFC7 640B$_H$	1(1L)	8
E7AP1M	ECC Encode / Decode Data Test Register	E710TED	0000 0000$_H$	32	FFC7 640C$_H$	1(1L)	32
E7AP1M	ECC Error Address Register	E710EAD	0000 0000$_H$	32	FFC7 6410$_H$	1(1L)	32
E7AP1C	ECC Control Register	E710CTL	001X$_H$	16	FFC7 6600$_H$	1(1L)	8, 16
E7AP1C	ECC Test Mode Control Register	E710TMC	0000_H	16	FFC7 6604$_H$	1(1L)	8, 16
E7AP1C	ECC Redundant Bit Data Control Test Register	E710TRC	0000 0000$_H$	32	FFC7 6608$_H$	1(1L)	32
E7AP1C	ECC Redundant Bit input and Output Substitution Buffer Register	ECERDB	00_H	8	FFC7 6608$_H$	1(1L)	8
E7AP1C	ECC Encode Test Register	ECECRD	00_H	8	FFC7 6609$_H$	1(1L)	8
E7AP1C	ECC7 Bit Redundant Bit Data Retention Test Register	ECHORD	00_H	8	FFC7 660A$_H$	1(1L)	8
E7AP1C	ECC Decode Syndrome Data Register	ECSYND	00_H	8	FFC7 660B$_H$	1(1L)	8
E7AP1C	ECC Encode / Decode Data Test Register	E710TED	0000 0000$_H$	32	FFC7 660C$_H$	1(1L)	32
E7AP1C	ECC Error Address Register	E710EAD	0000 0000$_H$	32	FFC7 6610$_H$	1(1L)	32
E7AP2M	ECC Control Register	E710CTL	001X$_H$	16	FFC7 6800$_H$	1(1L)	8, 16
E7AP2M	ECC Test Mode Control Register	E710TMC	0000_H	16	FFC7 6804$_H$	1(1L)	8, 16
E7AP2M	ECC Redundant Bit Data Control Test Register	E710TRC	0000 0000$_H$	32	FFC7 6808$_H$	1(1L)	32
E7AP2M	ECC Redundant Bit input and Output Substitution Buffer Register	ECERDB	00_H	8	FFC7 6808$_H$	1(1L)	8
E7AP2M	ECC Encode Test Register	ECECRD	00_H	8	FFC7 6809$_H$	1(1L)	8
E7AP2M	ECC7 Bit Redundant Bit Data Retention Test Register	ECHORD	00_H	8	FFC7 680A$_H$	1(1L)	8
E7AP2M	ECC Decode Syndrome Data Register	ECSYND	00_H	8	FFC7 680B$_H$	1(1L)	8
E7AP2M	ECC Encode / Decode Data Test Register	E710TED	0000 0000$_H$	32	FFC7 680C$_H$	1(1L)	32
E7AP2M	ECC Error Address Register	E710EAD	0000 0000$_H$	32	FFC7 6810$_H$	1(1L)	32
E7AP2C	ECC Control Register	E710CTL	001X$_H$	16	FFC7 6A00$_H$	1(1L)	8, 16
E7AP2C	ECC Test Mode Control Register	E710TMC	0000_H	16	FFC7 6A04$_H$	1(1L)	8, 16
E7AP2C	ECC Redundant Bit Data Control Test Register	E710TRC	0000 0000$_H$	32	FFC7 6A08$_H$	1(1L)	32
E7AP2C	ECC Redundant Bit input and Output Substitution Buffer Register	ECERDB	00_H	8	FFC7 6A08$_H$	1(1L)	8
E7AP2C	ECC Encode Test Register	ECECRD	00_H	8	FFC7 6A09$_H$	1(1L)	8
E7AP2C	ECC7 Bit Redundant Bit Data Retention Test Register	ECHORD	00_H	8	FFC7 6A0A$_H$	1(1L)	8
E7AP2C	ECC Decode Syndrome Data Register	ECSYND	00_H	8	FFC7 6A0B$_H$	1(1L)	8
E7AP2C	ECC Encode / Decode Data Test Register	E710TED	0000 0000$_H$	32	FFC7 6A0C$_H$	1(1L)	32
E7AP2C	ECC Error Address Register	E710EAD	0000 0000$_H$	32	FFC7 6A10$_H$	1(1L)	32
E7DF1M	ECC Control Register	E710CTL	001X$_H$	16	FFC7 D000$_H$	3	8, 16
E7DF1M	ECC Test Mode Control Register	E710TMC	0000_H	16	FFC7 D004$_H$	3	8, 16
E7DF1M	ECC Redundant Bit Data Control Test Register	E710TRC	0000 0000$_H$	32	FFC7 D008$_H$	3	32
E7DF1M	ECC Redundant Bit input and Output Substitution Buffer Register	ECERDB	00_H	8	FFC7 D008$_H$	3	8
E7DF1M	ECC Encode Test Register	ECECRD	00_H	8	FFC7 D009$_H$	3	8
E7DF1M	ECC7 Bit Redundant Bit Data Retention Test Register	ECHORD	00_H	8	FFC7 D00A$_H$	3	8
E7DF1M	ECC Decode Syndrome Data Register	ECSYND	00_H	8	FFC7 D00B$_H$	3	8

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
E7DF1M	ECC Encode / Decode Data Test Register	E710TED	0000 0000$_H$	32	FFC7 D00C$_H$	3	32
E7DF1M	ECC Error Address Register	E710EAD	0000 0000$_H$	32	FFC7 D010$_H$	3	32
E7DF1C	ECC Control Register	E710CTL	001X$_H$	16	FFC7 D200$_H$	3	8, 16
E7DF1C	ECC Test Mode Control Register	E710TMC	0000$_H$	16	FFC7 D204$_H$	3	8, 16
E7DF1C	ECC Redundant Bit Data Control Test Register	E710TRC	0000 0000$_H$	32	FFC7 D208$_H$	3	32
E7DF1C	ECC Redundant Bit input and Output Substitution Buffer Register	ECERDB	00$_H$	8	FFC7 D208$_H$	3	8
E7DF1C	ECC Encode Test Register	ECECRD	00$_H$	8	FFC7 D209$_H$	3	8
E7DF1C	ECC7 Bit Redundant Bit Data Retention Test Register	ECHORD	00$_H$	8	FFC7 D20A$_H$	3	8
E7DF1C	ECC Decode Syndrome Data Register	ECSYND	00$_H$	8	FFC7 D20B$_H$	3	8
E7DF1C	ECC Encode / Decode Data Test Register	E710TED	0000 0000$_H$	32	FFC7 D20C$_H$	3	32
E7DF1C	ECC Error Address Register	E710EAD	0000 0000$_H$	32	FFC7 D210$_H$	3	32
E6DF0M	ECC Control Register	E610CTL	001X$_H$	16	FFC7 D400$_H$	3	8, 16
E6DF0M	ECC Test Mode Control Register	E610TMC	0000$_H$	16	FFC7 D404$_H$	3	8, 16
E6DF0M	ECC Redundant Bit Data Control Test Register	E610TRC	0000 0000$_H$	32	FFC7 D408$_H$	3	32
E6DF0M	ECC Redundant Bit Output Data Monitor Register	ECRODM	00$_H$	8	FFC7 D408$_H$	3	8
E6DF0M	ECC Encode Test Register	ECECRD	00$_H$	8	FFC7 D409$_H$	3	8
E6DF0M	ECC6 Bit Redundant Bit Data Retention Test Register	ECHORD	00$_H$	8	FFC7 D40A$_H$	3	8
E6DF0M	ECC Decode Syndrome Data Register	ECSYND	00$_H$	8	FFC7 D40B$_H$	3	8
E6DF0M	ECC Encode / Decode Data Test Register	E610TED	0000 0000$_H$	32	FFC7 D40C$_H$	3	32
E6DF0M	ECC Redundant Bit input and Output Substitution Buffer Register	ECERDB	00$_H$	8	FFC7 D40C$_H$	3	8
E6DF0M	ECC Redundant Bit Input Data Monitor Register	ECRIDM	00$_H$	8	FFC7 D40D$_H$	3	8
E6DF0M	ECC Encode / Decode Data Input and Output Substitution Buffer Register	ECEDB	0000$_H$	16	FFC7 D40E$_H$	3	16
E6DF0M	ECC Error Address Register	E610EAD	0000 0000$_H$	32	FFC7 D410$_H$	3	32
E6DF0C	ECC Control Register	E610CTL	001X$_H$	16	FFC7 D600$_H$	3	8, 16
E6DF0C	ECC Test Mode Control Register	E610TMC	0000$_H$	16	FFC7 D604$_H$	3	8, 16
E6DF0C	ECC Redundant Bit Data Control Test Register	E610TRC	0000 0000$_H$	32	FFC7 D608$_H$	3	32
E6DF0C	ECC Redundant Bit Output Data Monitor Register	ECRODM	00$_H$	8	FFC7 D608$_H$	3	8
E6DF0C	ECC Encode Test Register	ECECRD	00$_H$	8	FFC7 D609$_H$	3	8
E6DF0C	ECC6 Bit Redundant Bit Data Retention Test Register	ECHORD	00$_H$	8	FFC7 D60A$_H$	3	8
E6DF0C	ECC Decode Syndrome Data Register	ECSYND	00$_H$	8	FFC7 D60B$_H$	3	8
E6DF0C	ECC Encode / Decode Data Test Register	E610TED	0000 0000$_H$	32	FFC7 D60C$_H$	3	32
E6DF0C	ECC Redundant Bit input and Output Substitution Buffer Register	ECERDB	00$_H$	8	FFC7 D60C$_H$	3	8
E6DF0C	ECC Redundant Bit Input Data Monitor Register	ECRIDM	00$_H$	8	FFC7 D60D$_H$	3	8
E6DF0C	ECC Encode / Decode Data Input and Output Substitution Buffer Register	ECEDB	0000$_H$	16	FFC7 D60E$_H$	3	16
E6DF0C	ECC Error Address Register	E610EAD	0000 0000$_H$	32	FFC7 D610$_H$	3	32
E6DF1M	ECC Control Register	E610CTL	001X$_H$	16	FFC7 D800$_H$	3	8, 16

(34/201)

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
E6DF1M	ECC Test Mode Control Register	E610TMC	0000_H	16	FFC7 $D804_H$	3	8, 16
E6DF1M	ECC Redundant Bit Data Control Test Register	E610TRC	0000 0000_H	32	FFC7 $D808_H$	3	32
E6DF1M	ECC Redundant Bit Output Data Monitor Register	ECRODM	00_H	8	FFC7 $D808_H$	3	8
E6DF1M	ECC Encode Test Register	ECECRD	00_H	8	FFC7 $D809_H$	3	8
E6DF1M	ECC6 Bit Redundant Bit Data Retention Test Register	ECHORD	00_H	8	FFC7 $D80A_H$	3	8
E6DF1M	ECC Decode Syndrome Data Register	ECSYND	00_H	8	FFC7 $D80B_H$	3	8
E6DF1M	ECC Encode / Decode Data Test Register	E610TED	0000 0000_H	32	FFC7 $D80C_H$	3	32
E6DF1M	ECC Redundant Bit input and Output Substitution Buffer Register	ECERDB	00_H	8	FFC7 $D80C_H$	3	8
E6DF1M	ECC Redundant Bit Input Data Monitor Register	ECRIDM	00_H	8	FFC7 $D80D_H$	3	8
E6DF1M	ECC Encode / Decode Data Input and Output Substitution Buffer Register	ECEDB	0000_H	16	FFC7 $D80E_H$	3	16
E6DF1M	ECC Error Address Register	E610EAD	0000 0000_H	32	FFC7 $D810_H$	3	32
E6DF1C	ECC Control Register	E610CTL	$001X_H$	16	FFC7 $DA00_H$	3	8, 16
E6DF1C	ECC Test Mode Control Register	E610TMC	0000_H	16	FFC7 $DA04_H$	3	8, 16
E6DF1C	ECC Redundant Bit Data Control Test Register	E610TRC	0000 0000_H	32	FFC7 $DA08_H$	3	32
E6DF1C	ECC Redundant Bit Output Data Monitoring Register	ECRODM	00_H	8	FFC7 $DA08_H$	3	8
E6DF1C	ECC Encode Test Register	ECECRD	00_H	8	FFC7 $DA09_H$	3	8
E6DF1C	ECC6 Bit Redundant Bit Data Retention Test Register	ECHORD	00_H	8	FFC7 $DA0A_H$	3	8
E6DF1C	ECC Decode Syndrome Data Register	ECSYND	00_H	8	FFC7 $DA0B_H$	3	8
E6DF1C	ECC Encode / Decode Data Test Register	E610TED	0000 0000_H	32	FFC7 $DA0C_H$	3	32
E6DF1C	ECC Redundant Bit Input and Output Substitution Buffer Register	ECERDB	00_H	8	FFC7 $DA0C_H$	3	8
E6DF1C	ECC Redundant Bit Input Data Monitoring Register	ECRIDM	00_H	8	FFC7 $DA0D_H$	3	8
E6DF1C	ECC Encode / Decode Data Input and Output Substitution Buffer Register	ECEDB	0000_H	16	FFC7 $DA0E_H$	3	16
E6DF1C	ECC Error Address Register	E610EAD	0000 0000_H	32	FFC7 $DA10_H$	3	32
APDP	P-Bus Data Parity Status Register PT0A	APDPERRST_PT0A	0000 0000_H	32	FFC8 5000_H	3	8, 16, 32
APDP	P-Bus Data Parity Status Clear Register PT0A	APDPERRSTC_PT0A	0000 0000_H	32	FFC8 5004_H	3	8, 16, 32
APDP	P-Bus Data Parity Test Mode Control Register PT0A	APDPTMC_PT0A	0000 0000_H	32	FFC8 5008_H	3	16, 32
APDP	P-Bus Data Parity Error Address Register PT0A	APDPERRADR_PT0A	0000 0000_H	32	FFC8 $500C_H$	3	32
APDP	P-Bus Data Parity Status Register PT0B	APDPERRST_PT0B	0000 0000_H	32	FFC8 5020_H	3	8, 16, 32
APDP	P-Bus Data Parity Status Clear Register PT0B	APDPERRSTC_PT0B	0000 0000_H	32	FFC8 5024_H	3	8, 16, 32
APDP	P-Bus Data Parity Test Mode Control Register PT0B	APDPTMC_PT0B	0000 0000_H	32	FFC8 5028_H	3	16, 32
APDP	P-Bus Data Parity Error Address Register PT0B	APDPERRADR_PT0B	0000 0000_H	32	FFC8 $502C_H$	3	32
APDP	P-Bus Data Parity Status Register PT1A	APDPERRST_PT1A	0000 0000_H	32	FFC8 5040_H	3	8, 16, 32
APDP	P-Bus Data Parity Status Clear Register PT1A	APDPERRSTC_PT1A	0000 0000_H	32	FFC8 5044_H	3	8, 16, 32
APDP	P-Bus Data Parity Test Mode Control Register PT1A	APDPTMC_PT1A	0000 0000_H	32	FFC8 5048_H	3	16, 32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
APDP	P-Bus Data Parity Error Address Register PT1A	APDPERRADR_PT1A	0000 0000$_H$	32	FFC8 504C$_H$	3	32
APDP	P-Bus Data Parity Status Register PT1B	APDPERRST_PT1B	0000 0000$_H$	32	FFC8 5060$_H$	3	8, 16, 32
APDP	P-Bus Data Parity Status Clear Register PT1B	APDPERRSTC_PT1B	0000 0000$_H$	32	FFC8 5064$_H$	3	8, 16, 32
APDP	P-Bus Data Parity Test Mode Control Register PT1B	APDPTMC_PT1B	0000 0000$_H$	32	FFC8 5068$_H$	3	16, 32
APDP	P-Bus Data Parity Error Address Register PT1B	APDPERRADR_PT1B	0000 0000$_H$	32	FFC8 506C$_H$	3	32
APDP	P-Bus Data Parity Status Register DNF0	APDPERRST_DNF0	0000 0000$_H$	32	FFC8 50C0$_H$	3	8, 16, 32
APDP	P-Bus Data Parity Status Clear Register DNF0	APDPERRSTC_DNF0	0000 0000$_H$	32	FFC8 50C4$_H$	3	8, 16, 32
APDP	P-Bus Data Parity Test Mode Control Register DNF0	APDPTMC_DNF0	0000 0000$_H$	32	FFC8 50C8$_H$	3	16, 32
APDP	P-Bus Data Parity Error Address Register DNF0	APDPERRADR_DNF0	0000 0000$_H$	32	FFC8 50CC$_H$	3	32
APDP	P-Bus Data Parity Status Register DNF1	APDPERRST_DNF1	0000 0000$_H$	32	FFC8 50E0$_H$	3	8, 16, 32
APDP	P-Bus Data Parity Status Clear Register DNF1	APDPERRSTC_DNF1	0000 0000$_H$	32	FFC8 50E4$_H$	3	8, 16, 32
APDP	P-Bus Data Parity Test Mode Control Register DNF1	APDPTMC_DNF1	0000 0000$_H$	32	FFC8 50E8$_H$	3	16, 32
APDP	P-Bus Data Parity Error Address Register DNF1	APDPERRADR_DNF1	0000 0000$_H$	32	FFC8 50EC$_H$	3	32
APDP	P-Bus Data Parity Status Register DNF2	APDPERRST_DNF2	0000 0000$_H$	32	FFC8 5800$_H$	3	8, 16, 32
APDP	P-Bus Data Parity Status Clear Register DNF2	APDPERRSTC_DNF2	0000 0000$_H$	32	FFC8 5804$_H$	3	8, 16, 32
APDP	P-Bus Data Parity Test Mode Control Register DNF2	APDPTMC_DNF2	0000 0000$_H$	32	FFC8 5808$_H$	3	16, 32
APDP	P-Bus Data Parity Error Address Register DNF2	APDPERRADR_DNF2	0000 0000$_H$	32	FFC8 580C$_H$	3	32
APDP	P-Bus Data Parity Status Register DNF3	APDPERRST_DNF3	0000 0000$_H$	32	FFC8 5820$_H$	3	8, 16, 32
APDP	P-Bus Data Parity Status Clear Register DNF3	APDPERRSTC_DNF3	0000 0000$_H$	32	FFC8 5824$_H$	3	8, 16, 32
APDP	P-Bus Data Parity Test Mode Control Register DNF3	APDPTMC_DNF3	0000 0000$_H$	32	FFC8 5828$_H$	3	16, 32
APDP	P-Bus Data Parity Error Address Register DNF3	APDPERRADR_DNF3	0000 0000$_H$	32	FFC8 582C$_H$	3	32
APDP	P-Bus Data Parity Status Register DNF4	APDPERRST_DNF4	0000 0000$_H$	32	FFC8 5840$_H$	3	8, 16, 32
APDP	P-Bus Data Parity Status Clear Register DNF4	APDPERRSTC_DNF4	0000 0000$_H$	32	FFC8 5844$_H$	3	8, 16, 32
APDP	P-Bus Data Parity Test Mode Control Register DNF4	APDPTMC_DNF4	0000 0000$_H$	32	FFC8 5848$_H$	3	16, 32
APDP	P-Bus Data Parity Error Address Register DNF4	APDPERRADR_DNF4	0000 0000$_H$	32	FFC8 584C$_H$	3	32
APDP	P-Bus Data Parity Status Register DNF5	APDPERRST_DNF5	0000 0000$_H$	32	FFC8 5860$_H$	3	8, 16, 32
APDP	P-Bus Data Parity Status Clear Register DNF5	APDPERRSTC_DNF5	0000 0000$_H$	32	FFC8 5864$_H$	3	8, 16, 32
APDP	P-Bus Data Parity Test Mode Control Register DNF5	APDPTMC_DNF5	0000 0000$_H$	32	FFC8 5868$_H$	3	16, 32
APDP	P-Bus Data Parity Error Address Register DNF5	APDPERRADR_DNF5	0000 0000$_H$	32	FFC8 586C$_H$	3	32
APDP	P-Bus Data Parity Status Register DNF6	APDPERRST_DNF6	0000 0000$_H$	32	FFC8 5880$_H$	3	8, 16, 32
APDP	P-Bus Data Parity Status Clear Register DNF6	APDPERRSTC_DNF6	0000 0000$_H$	32	FFC8 5884$_H$	3	8, 16, 32
APDP	P-Bus Data Parity Test Mode Control Register DNF6	APDPTMC_DNF6	0000 0000$_H$	32	FFC8 5888$_H$	3	16, 32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
APDP	P-Bus Data Parity Error Address Register DNF6	APDPERRADR_DNF6	0000 0000$_H$	32	FFC8 588C$_H$	3	32
APDP	P-Bus Data Parity Status Register DNF7	APDPERRST_DNF7	0000 0000$_H$	32	FFC8 58A0$_H$	3	8, 16, 32
APDP	P-Bus Data Parity Status Clear Register DNF7	APDPERRSTC_DNF7	0000 0000$_H$	32	FFC8 58A4$_H$	3	8, 16, 32
APDP	P-Bus Data Parity Test Mode Control Register DNF7	APDPTMC_DNF7	0000 0000$_H$	32	FFC8 58A8$_H$	3	16, 32
APDP	P-Bus Data Parity Error Address Register DNF7	APDPERRADR_DNF7	0000 0000$_H$	32	FFC8 58AC$_H$	3	32
APDP	P-Bus Data Parity Status Register DADC	APDPERRST_DADC	0000 0000$_H$	32	FFC8 7000$_H$	3	8, 16, 32
APDP	P-Bus Data Parity Status Clear Register DADC	APDPERRSTC_DADC	0000 0000$_H$	32	FFC8 7004$_H$	3	8, 16, 32
APDP	P-Bus Data Parity Test Mode Control Register DADC	APDPTMC_DADC	0000 0000$_H$	32	FFC8 7008$_H$	3	16, 32
APDP	P-Bus Data Parity Error Address Register DADC	APDPERRADR_DADC	0000 0000$_H$	32	FFC8 700C$_H$	3	32
APDP	P-Bus Data Parity Status Register DAD0	APDPERRST_DAD0	0000 0000$_H$	32	FFC8 7020$_H$	3	8, 16, 32
APDP	P-Bus Data Parity Status Clear Register DAD0	APDPERRSTC_DAD0	0000 0000$_H$	32	FFC8 7024$_H$	3	8, 16, 32
APDP	P-Bus Data Parity Test Mode Control Register DAD0	APDPTMC_DAD0	0000 0000$_H$	32	FFC8 7028$_H$	3	16, 32
APDP	P-Bus Data Parity Error Address Register DAD0	APDPERRADR_DAD0	0000 0000$_H$	32	FFC8 702C$_H$	3	32
APDP	P-Bus Data Parity Status Register DAD1	APDPERRST_DAD1	0000 0000$_H$	32	FFC8 7040$_H$	3	8, 16, 32
APDP	P-Bus Data Parity Status Clear Register DAD1	APDPERRSTC_DAD1	0000 0000$_H$	32	FFC8 7044$_H$	3	8, 16, 32
APDP	P-Bus Data Parity Test Mode Control Register DAD1	APDPTMC_DAD1	0000 0000$_H$	32	FFC8 7048$_H$	3	16, 32
APDP	P-Bus Data Parity Error Address Register DAD1	APDPERRADR_DAD1	0000 0000$_H$	32	FFC8 704C$_H$	3	32
APDP	P-Bus Data Parity Status Register DAD2	APDPERRST_DAD2	0000 0000$_H$	32	FFC8 7060$_H$	3	8, 16, 32
APDP	P-Bus Data Parity Status Clear Register DAD2	APDPERRSTC_DAD2	0000 0000$_H$	32	FFC8 7064$_H$	3	8, 16, 32
APDP	P-Bus Data Parity Test Mode Control Register DAD2	APDPTMC_DAD2	0000 0000$_H$	32	FFC8 7068$_H$	3	16, 32
APDP	P-Bus Data Parity Error Address Register DAD2	APDPERRADR_DAD2	0000 0000$_H$	32	FFC8 706C$_H$	3	32
APDP	P-Bus Data Parity Status Register DAD3	APDPERRST_DAD3	0000 0000$_H$	32	FFC8 7080$_H$	3	8, 16, 32
APDP	P-Bus Data Parity Status Clear Register DAD3	APDPERRSTC_DAD3	0000 0000$_H$	32	FFC8 7084$_H$	3	8, 16, 32
APDP	P-Bus Data Parity Test Mode Control Register DAD3	APDPTMC_DAD3	0000 0000$_H$	32	FFC8 7088$_H$	3	16, 32
APDP	P-Bus Data Parity Error Address Register DAD3	APDPERRADR_DAD3	0000 0000$_H$	32	FFC8 708C$_H$	3	32
APDP	P-Bus Data Parity Status Register DAD4	APDPERRST_DAD4	0000 0000$_H$	32	FFC8 70A0$_H$	3	8, 16, 32
APDP	P-Bus Data Parity Status Clear Register DAD4	APDPERRSTC_DAD4	0000 0000$_H$	32	FFC8 70A4$_H$	3	8, 16, 32
APDP	P-Bus Data Parity Test Mode Control Register DAD4	APDPTMC_DAD4	0000 0000$_H$	32	FFC8 70A8$_H$	3	16, 32
APDP	P-Bus Data Parity Error Address Register DAD4	APDPERRADR_DAD4	0000 0000$_H$	32	FFC8 70AC$_H$	3	32
APDP	P-Bus Data Parity Status Register DAD5	APDPERRST_DAD5	0000 0000$_H$	32	FFC8 70C0$_H$	3	8, 16, 32
APDP	P-Bus Data Parity Status Clear Register DAD5	APDPERRSTC_DAD5	0000 0000$_H$	32	FFC8 70C4$_H$	3	8, 16, 32
APDP	P-Bus Data Parity Test Mode Control Register DAD5	APDPTMC_DAD5	0000 0000$_H$	32	FFC8 70C8$_H$	3	16, 32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
APDP	P-Bus Data Parity Error Address Register DAD5	APDPERRADR_DAD5	0000 0000$_H$	32	FFC8 70CC$_H$	3	32
APDP	P-Bus Data Parity Status Register DAD6	APDPERRST_DAD6	0000 0000$_H$	32	FFC8 70E0$_H$	3	8, 16, 32
APDP	P-Bus Data Parity Status Clear Register DAD6	APDPERRSTC_DAD6	0000 0000$_H$	32	FFC8 70E4$_H$	3	8, 16, 32
APDP	P-Bus Data Parity Test Mode Control Register DAD6	APDPTMC_DAD6	0000 0000$_H$	32	FFC8 70E8$_H$	3	16, 32
APDP	P-Bus Data Parity Error Address Register DAD6	APDPERRADR_DAD6	0000 0000$_H$	32	FFC8 70EC$_H$	3	32
APDP	P-Bus Data Parity Status Register DAD7	APDPERRST_DAD7	0000 0000$_H$	32	FFC8 7100$_H$	3	8, 16, 32
APDP	P-Bus Data Parity Status Clear Register DAD7	APDPERRSTC_DAD7	0000 0000$_H$	32	FFC8 7104$_H$	3	8, 16, 32
APDP	P-Bus Data Parity Test Mode Control Register DAD7	APDPTMC_DAD7	0000 0000$_H$	32	FFC8 7108$_H$	3	16, 32
APDP	P-Bus Data Parity Error Address Register DAD7	APDPERRADR_DAD7	0000 0000$_H$	32	FFC8 710C$_H$	3	32
APDP	P-Bus Data Parity Status Register ADC0	APDPERRST_ADC0	0000 0000$_H$	32	FFC8 8000$_H$	3	8, 16, 32
APDP	P-Bus Data Parity Status Clear Register ADC0	APDPERRSTC_ADC0	0000 0000$_H$	32	FFC8 8004$_H$	3	8, 16, 32
APDP	P-Bus Data Parity Test Mode Control Register ADC0	APDPTMC_ADC0	0000 0000$_H$	32	FFC8 8008$_H$	3	16, 32
APDP	P-Bus Data Parity Error Address Register ADC0	APDPERRADR_ADC0	0000 0000$_H$	32	FFC8 800C$_H$	3	32
APDP	P-Bus Data Parity Status Register ADC1	APDPERRST_ADC1	0000 0000$_H$	32	FFC8 8020$_H$	3	8, 16, 32
APDP	P-Bus Data Parity Status Clear Register ADC1	APDPERRSTC_ADC1	0000 0000$_H$	32	FFC8 8024$_H$	3	8, 16, 32
APDP	P-Bus Data Parity Test Mode Control Register ADC1	APDPTMC_ADC1	0000 0000$_H$	32	FFC8 8028$_H$	3	16, 32
APDP	P-Bus Data Parity Error Address Register ADC1	APDPERRADR_ADC1	0000 0000$_H$	32	FFC8 802C$_H$	3	32
APDP	P-Bus Data Parity Status Register IFC	APDPERRST_IFC	0000 0000$_H$	32	FFC8 8100$_H$	3	8, 16, 32
APDP	P-Bus Data Parity Status Clear Register IFC	APDPERRSTC_IFC	0000 0000$_H$	32	FFC8 8104$_H$	3	8, 16, 32
APDP	P-Bus Data Parity Test Mode Control Register IFC	APDPTMC_IFC	0000 0000$_H$	32	FFC8 8108$_H$	3	16, 32
APDP	P-Bus Data Parity Error Address Register IFC	APDPERRADR_IFC	0000 0000$_H$	32	FFC8 810C$_H$	3	32
APDP	P-Bus Data Parity Status Register ASF	APDPERRST_ASF	0000 0000$_H$	32	FFC8 8120$_H$	3	8, 16, 32
APDP	P-Bus Data Parity Status Clear Register ASF	APDPERRSTC_ASF	0000 0000$_H$	32	FFC8 8124$_H$	3	8, 16, 32
APDP	P-Bus Data Parity Test Mode Control Register ASF	APDPTMC_ASF	0000 0000$_H$	32	FFC8 8128$_H$	3	16, 32
APDP	P-Bus Data Parity Error Address Register ASF	APDPERRADR_ASF	0000 0000$_H$	32	FFC8 812C$_H$	3	32
ECMM	ECM Master Error Set Trigger Register	ECMMESET	00$_H$	8	FFCB 0000$_H$	3	8
ECMM	ECM Master Error Clear Trigger Register	ECMMECLR	00$_H$	8	FFCB 0004$_H$	3	8
ECMM	ECM Master Error Source Status Register 0	ECMMESSTR0	0000 0000$_H$	32	FFCB 0008$_H$	3	32
ECMM	ECM Master Error Source Status Register 1	ECMMESSTR1	0000 0000$_H$	32	FFCB 000C$_H$	3	32
ECMM	ECM Master Protection Command Register	ECMMPCMD0	XXXX XXXX$_H$	32	FFCB 0010$_H$	3	32
ECMC	ECM Checker Error Set Trigger Register	ECMCESET	00$_H$	8	FFCB 1000$_H$	3	8
ECMC	ECM Checker Error Clear Trigger Register	ECMCECLR	00$_H$	8	FFCB 1004$_H$	3	8

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
ECMC	ECM Checker Error Source Status Register 0	ECMCESSTR0	0000 0000_H	32	FFCB 1008_H	3	32
ECMC	ECM Checker Error Source Status Register 1	ECMCESSTR1	0000 0000_H	32	FFCB $100C_H$	3	32
ECMC	ECM Checker Protection Command Register	ECMCPCMD0	XXXX $XXXX_H$	32	FFCB 1010_H	3	32
ECM	ECM Error Pulse Configuration Register	ECMEPCFG	00_H	8	FFCB 2000_H	3	8
ECM	ECM Maskable Interrupt Configuration Register 0	ECMMICFG0	0000 0000_H	32	FFCB 2004_H	3	32
ECM	ECM Maskable Interrupt Configuration Register 1	ECMMICFG1	0000 0000_H	32	FFCB 2008_H	3	32
ECM	ECM FE Level Interrupt Configuration Register 0	ECMNMICFG0	0000 0000_H	32	FFCB $200C_H$	3	32
ECM	ECM FE Level Interrupt Configuration Register 1	ECMNMICFG1	0000 0000_H	32	FFCB 2010_H	3	32
ECM	ECM Internal Reset Configuration Register 0	ECMIRCFG0	0000 $000F_H$	32	FFCB 2014_H	3	32
ECM	ECM Internal Reset Configuration Register 1	ECMIRCFG1	0000 0000_H	32	FFCB 2018_H	3	32
ECM	ECM Error Mask Register 0	ECMEMK0	0000 0000_H	32	FFCB $201C_H$	3	32
ECM	ECM Error Mask Register 1	ECMEMK1	0000 0000_H	32	FFCB 2020_H	3	32
ECM	ECM Error Source Status Clear Register 0	ECMESSTC0	0000 0000_H	32	FFCB 2024_H	3	32
ECM	ECM Error Source Status Clear Register 1	ECMESSTC1	0000 0000_H	32	FFCB 2028_H	3	32
ECM	ECM Protection Command Register	ECMPCMD1	XXXX $XXXX_H$	32	FFCB $202C_H$	3	32
ECM	ECM Protection Status Register	ECMPS	00_H	8	FFCB 2030_H	3	8
ECM	ECM Pseudo Error Trigger Register 0	ECMPE0	0000 0000_H	32	FFCB 2034_H	3	32
ECM	ECM Pseudo Error Trigger Register 1	ECMPE1	0000 0000_H	32	FFCB 2038_H	3	32
ECM	ECM Delay Timer Control Register	ECMDTMCTL	00_H	8	FFCB $203C_H$	3	8
ECM	ECM Delay Timer Register	ECMDTMR	0000_H	16	FFCB 2040_H	3	16
ECM	ECM Delay Timer Comparison Register	ECMDTMCMP	0000_H	16	FFCB 2044_H	3	16
ECM	ECM Delay Timer Configuration Register 0	ECMDTMCFG0	0000 0000_H	32	FFCB 2048_H	3	32
ECM	ECM Delay Timer Configuration Register 1	ECMDTMCFG1	0000 0000_H	32	FFCB $204C_H$	3	32
ECM	ECM Delay Timer Configuration Register 2	ECMDTMCFG2	0000 0000_H	32	FFCB 2050_H	3	32
ECM	ECM Delay Timer Configuration Register 3	ECMDTMCFG3	0000 0000_H	32	FFCB 2054_H	3	32
FLASH	Product Name Storage Register 1	PRDNAME1	XXXX $XXXX_H$	32	FFCD $00D0_H$	3	32
FLASH	Product Name Storage Register 2	PRDNAME2	XXXX $XXXX_H$	32	FFCD $00D4_H$	3	32
FLASH	Product Name Storage Register 3	PRDNAME3	XXXX $XXXX_H$	32	FFCD $00D8_H$	3	32
FLASH	Product Name Storage Register 4	PRDNAME4	XXXX $XXXX_H$	32	FFCD $00DC_H$	3	32
RLN210G	LIN Wake-up Baud Rate Selection Register	LWBR	00_H	8	FFCE 0001_H	3	8
RLN210G	LIN Baud Rate Prescaler 0 Register	LBRP0	00_H	8	FFCE 0002_H	3	8
RLN210G	LIN Baud Rate Prescaler 1 Register	LBRP1	00_H	8	FFCE 0003_H	3	8
RLN210G	LIN Self Test Control Register	LSTC	00_H	8	FFCE 0004_H	3	8
RLN2100	LIN0 Mode Register	L0MD	00_H	8	FFCE 0008_H	3	8
RLN2100	LIN0 Break Filed Setting Register	L0BFC	00_H	8	FFCE 0009_H	3	8
RLN2100	LIN0 Space Setting Register	L0SC	00_H	8	FFCE $000A_H$	3	8
RLN2100	LIN0 Wake-up Setting Register	L0WUP	00_H	8	FFCE $000B_H$	3	8
RLN2100	LIN0 Interrupt Enable Register	L0IE	00_H	8	FFCE $000C_H$	3	8
RLN2100	LIN0 Error Detection Enable Register	L0EDE	00_H	8	FFCE $000D_H$	3	8

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
RLN2100	LIN0 Control Register	L0CUC	00_H	8	FFCE 000E$_H$	3	8
RLN2100	LIN0 Transmission Control Register	L0TRC	00_H	8	FFCE 0010$_H$	3	8
RLN2100	LIN0 Mode Status Register	L0MST	00_H	8	FFCE 0011$_H$	3	8
RLN2100	LIN0 Status Register	L0ST	00_H	8	FFCE 0012$_H$	3	8
RLN2100	LIN0 Error Status Register	L0EST	00_H	8	FFCE 0013$_H$	3	8
RLN2100	LIN0 Data Field Setting Register	L0DFC	00_H	8	FFCE 0014$_H$	3	8
RLN2100	LIN0 ID Buffer Register	L0IDB	XX_H	8	FFCE 0015$_H$	3	8
RLN2100	LIN0 Check Sum Buffer Register	L0CBR	XX_H	8	FFCE 0016$_H$	3	8
RLN2100	LIN0 Data Buffer 1 Register	L0DBR1	XX_H	8	FFCE 0018$_H$	3	8
RLN2100	LIN0 Data Buffer 2 Register	L0DBR2	XX_H	8	FFCE 0019$_H$	3	8
RLN2100	LIN0 Data Buffer 3 Register	L0DBR3	XX_H	8	FFCE 001A$_H$	3	8
RLN2100	LIN0 Data Buffer 4 Register	L0DBR4	XX_H	8	FFCE 001B$_H$	3	8
RLN2100	LIN0 Data Buffer 5 Register	L0DBR5	XX_H	8	FFCE 001C$_H$	3	8
RLN2100	LIN0 Data Buffer 6 Register	L0DBR6	XX_H	8	FFCE 001D$_H$	3	8
RLN2100	LIN0 Data Buffer 7 Register	L0DBR7	XX_H	8	FFCE 001E$_H$	3	8
RLN2100	LIN0 Data Buffer 8 Register	L0DBR8	XX_H	8	FFCE 001F$_H$	3	8
RSCAN0	Channel 0 Configuration Register	RSCAN0C0CFG	0000 0000$_H$	32	FFD0 0000$_H$	2	8, 16, 32
RSCAN0	Channel 0 Control Register	RSCAN0C0CTR	0000 0005$_H$	32	FFD0 0004$_H$	2	8, 16, 32
RSCAN0	Channel 0 Status Register	RSCAN0C0STS	0000 0005$_H$	32	FFD0 0008$_H$	2	8, 16, 32
RSCAN0	Channel 0 Error Flag Register	RSCAN0C0ERFL	0000 0000$_H$	32	FFD0 000C$_H$	2	8, 16, 32
RSCAN0	Channel 1 Configuration Register	RSCAN0C1CFG	0000 0000$_H$	32	FFD0 0010$_H$	2	8, 16, 32
RSCAN0	Channel 1 Control Register	RSCAN0C1CTR	0000 0005$_H$	32	FFD0 0014$_H$	2	8, 16, 32
RSCAN0	Channel 1 Status Register	RSCAN0C1STS	0000 0005$_H$	32	FFD0 0018$_H$	2	8, 16, 32
RSCAN0	Channel 1 Error Flag Register	RSCAN0C1ERFL	0000 0000$_H$	32	FFD0 001C$_H$	2	8, 16, 32
RSCAN0	Channel 2 Configuration Register	RSCAN0C2CFG	0000 0000$_H$	32	FFD0 0020$_H$	2	8, 16, 32
RSCAN0	Channel 2 Control Register	RSCAN0C2CTR	0000 0005$_H$	32	FFD0 0024$_H$	2	8, 16, 32
RSCAN0	Channel 2 Status Register	RSCAN0C2STS	0000 0005$_H$	32	FFD0 0028$_H$	2	8, 16, 32
RSCAN0	Channel 2 Error Flag Register	RSCAN0C2ERFL	0000 0000$_H$	32	FFD0 002C$_H$	2	8, 16, 32
RSCAN0	Channel 3 Configuration Register	RSCAN0C3CFG	0000 0000$_H$	32	FFD0 0030$_H$	2	8, 16, 32
RSCAN0	Channel 3 Control Register	RSCAN0C3CTR	0000 0005$_H$	32	FFD0 0034$_H$	2	8, 16, 32
RSCAN0	Channel 3 Status Register	RSCAN0C3STS	0000 0005$_H$	32	FFD0 0038$_H$	2	8, 16, 32
RSCAN0	Channel 3 Error Flag Register	RSCAN0C3ERFL	0000 0000$_H$	32	FFD0 003C$_H$	2	8, 16, 32
RSCAN0	Global Configuration Register	RSCAN0GCFG	0000 0000$_H$	32	FFD0 0084$_H$	2	8, 16, 32
RSCAN0	Global Control Register	RSCAN0GCTR	0000 0005$_H$	32	FFD0 0088$_H$	2	8, 16, 32
RSCAN0	Global Status Register	RSCAN0GSTS	0000 000D$_H$	32	FFD0 008C$_H$	2	8, 16, 32
RSCAN0	Global Error Flag Register	RSCAN0GERFL	0000 0000$_H$	32	FFD0 0090$_H$	2	8, 16, 32
RSCAN0	Global Time Stamp Counter Register	RSCAN0GTSC	0000 0000$_H$	32	FFD0 0094$_H$	2	16, 32
RSCAN0	Receive Rule Entry Control Register	RSCAN0GAFLECTR	0000 0000$_H$	32	FFD0 0098$_H$	2	8, 16, 32
RSCAN0	Receive Rule Configuration Register 0	RSCAN0GAFLCFG0	0000 0000$_H$	32	FFD0 009C$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Number Register	RSCAN0RMNB	0000 0000$_H$	32	FFD0 00A4$_H$	2	8, 16, 32
RSCAN0	Receive Buffer new Data Register 0	RSCAN0RMND0	0000 0000$_H$	32	FFD0 00A8$_H$	2	8, 16, 32
RSCAN0	Receive Buffer new Data Register 1	RSCAN0RMND1	0000 0000$_H$	32	FFD0 00AC$_H$	2	8, 16, 32
RSCAN0	Receive FIFO Buffer Configuration Control Register 0	RSCAN0RFCC0	0000 0000$_H$	32	FFD0 00B8$_H$	2	8, 16, 32
RSCAN0	Receive FIFO Buffer Configuration Control Register 1	RSCAN0RFCC1	0000 0000$_H$	32	FFD0 00BC$_H$	2	8, 16, 32
RSCAN0	Receive FIFO Buffer Configuration Control Register 2	RSCAN0RFCC2	0000 0000$_H$	32	FFD0 00C0$_H$	2	8, 16, 32

(40/201)

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
RSCAN0	Receive FIFO Buffer Configuration Control Register 3	RSCAN0RFCC3	0000 0000$_H$	32	FFD0 00C4$_H$	2	8, 16, 32
RSCAN0	Receive FIFO Buffer Configuration Control Register 4	RSCAN0RFCC4	0000 0000$_H$	32	FFD0 00C8$_H$	2	8, 16, 32
RSCAN0	Receive FIFO Buffer Configuration Control Register 5	RSCAN0RFCC5	0000 0000$_H$	32	FFD0 00CC$_H$	2	8, 16, 32
RSCAN0	Receive FIFO Buffer Configuration Control Register 6	RSCAN0RFCC6	0000 0000$_H$	32	FFD0 00D0$_H$	2	8, 16, 32
RSCAN0	Receive FIFO Buffer Configuration Control Register 7	RSCAN0RFCC7	0000 0000$_H$	32	FFD0 00D4$_H$	2	8, 16, 32
RSCAN0	Receive FIFO Buffer Status Register 0	RSCAN0RFSTS0	0000 0001$_H$	32	FFD0 00D8$_H$	2	8, 16, 32
RSCAN0	Receive FIFO Buffer Status Register 1	RSCAN0RFSTS1	0000 0001$_H$	32	FFD0 00DC$_H$	2	8, 16, 32
RSCAN0	Receive FIFO Buffer Status Register 2	RSCAN0RFSTS2	0000 0001$_H$	32	FFD0 00E0$_H$	2	8, 16, 32
RSCAN0	Receive FIFO Buffer Status Register 3	RSCAN0RFSTS3	0000 0001$_H$	32	FFD0 00E4$_H$	2	8, 16, 32
RSCAN0	Receive FIFO Buffer Status Register 4	RSCAN0RFSTS4	0000 0001$_H$	32	FFD0 00E8$_H$	2	8, 16, 32
RSCAN0	Receive FIFO Buffer Status Register 5	RSCAN0RFSTS5	0000 0001$_H$	32	FFD0 00EC$_H$	2	8, 16, 32
RSCAN0	Receive FIFO Buffer Status Register 6	RSCAN0RFSTS6	0000 0001$_H$	32	FFD0 00F0$_H$	2	8, 16, 32
RSCAN0	Receive FIFO Buffer Status Register 7	RSCAN0RFSTS7	0000 0001$_H$	32	FFD0 00F4$_H$	2	8, 16, 32
RSCAN0	Receive FIFO Buffer Pointer Control Register 0	RSCAN0RFPCTR0	0000 0000$_H$	32	FFD0 00F8$_H$	2	8, 16, 32
RSCAN0	Receive FIFO Buffer Pointer Control Register 1	RSCAN0RFPCTR1	0000 0000$_H$	32	FFD0 00FC$_H$	2	8, 16, 32
RSCAN0	Receive FIFO Buffer Pointer Control Register 2	RSCAN0RFPCTR2	0000 0000$_H$	32	FFD0 0100$_H$	2	8, 16, 32
RSCAN0	Receive FIFO Buffer Pointer Control Register 3	RSCAN0RFPCTR3	0000 0000$_H$	32	FFD0 0104$_H$	2	8, 16, 32
RSCAN0	Receive FIFO Buffer Pointer Control Register 4	RSCAN0RFPCTR4	0000 0000$_H$	32	FFD0 0108$_H$	2	8, 16, 32
RSCAN0	Receive FIFO Buffer Pointer Control Register 5	RSCAN0RFPCTR5	0000 0000$_H$	32	FFD0 010C$_H$	2	8, 16, 32
RSCAN0	Receive FIFO Buffer Pointer Control Register 6	RSCAN0RFPCTR6	0000 0000$_H$	32	FFD0 0110$_H$	2	8, 16, 32
RSCAN0	Receive FIFO Buffer Pointer Control Register 7	RSCAN0RFPCTR7	0000 0000$_H$	32	FFD0 0114$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Configuration Control Register 0	RSCAN0CFCC0	0000 0000$_H$	32	FFD0 0118$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Configuration Control Register 1	RSCAN0CFCC1	0000 0000$_H$	32	FFD0 011C$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Configuration Control Register 2	RSCAN0CFCC2	0000 0000$_H$	32	FFD0 0120$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Configuration Control Register 3	RSCAN0CFCC3	0000 0000$_H$	32	FFD0 0124$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Configuration Control Register 4	RSCAN0CFCC4	0000 0000$_H$	32	FFD0 0128$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Configuration Control Register 5	RSCAN0CFCC5	0000 0000$_H$	32	FFD0 012C$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Configuration Control Register 6	RSCAN0CFCC6	0000 0000$_H$	32	FFD0 0130$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Configuration Control Register 7	RSCAN0CFCC7	0000 0000$_H$	32	FFD0 0134$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Configuration Control Register 8	RSCAN0CFCC8	0000 0000$_H$	32	FFD0 0138$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Configuration Control Register 9	RSCAN0CFCC9	0000 0000$_H$	32	FFD0 013C$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Configuration Control Register 10	RSCAN0CFCC10	0000 0000$_H$	32	FFD0 0140$_H$	2	8, 16, 32

(41/201)

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
RSCAN0	Transmit/receive FIFO Buffer Configuration Control Register 11	RSCAN0CFCC11	0000 0000$_H$	32	FFD0 0144$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Status Register 0	RSCAN0CFSTS0	0000 0001$_H$	32	FFD0 0178$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Status Register 1	RSCAN0CFSTS1	0000 0001$_H$	32	FFD0 017C$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Status Register 2	RSCAN0CFSTS2	0000 0001$_H$	32	FFD0 0180$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Status Register 3	RSCAN0CFSTS3	0000 0001$_H$	32	FFD0 0184$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Status Register 4	RSCAN0CFSTS4	0000 0001$_H$	32	FFD0 0188$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Status Register 5	RSCAN0CFSTS5	0000 0001$_H$	32	FFD0 018C$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Status Register 6	RSCAN0CFSTS6	0000 0001$_H$	32	FFD0 0190$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Status Register 7	RSCAN0CFSTS7	0000 0001$_H$	32	FFD0 0194$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Status Register 8	RSCAN0CFSTS8	0000 0001$_H$	32	FFD0 0198$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Status Register 9	RSCAN0CFSTS9	0000 0001$_H$	32	FFD0 019C$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Status Register 10	RSCAN0CFSTS10	0000 0001$_H$	32	FFD0 01A0$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Status Register 11	RSCAN0CFSTS11	0000 0001$_H$	32	FFD0 01A4$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Pointer Control Register 0	RSCAN0CFPCTR0	0000 0000$_H$	32	FFD0 01D8$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Pointer Control Register 1	RSCAN0CFPCTR1	0000 0000$_H$	32	FFD0 01DC$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Pointer Control Register 2	RSCAN0CFPCTR2	0000 0000$_H$	32	FFD0 01E0$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Pointer Control Register 3	RSCAN0CFPCTR3	0000 0000$_H$	32	FFD0 01E4$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Pointer Control Register 4	RSCAN0CFPCTR4	0000 0000$_H$	32	FFD0 01E8$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Pointer Control Register 5	RSCAN0CFPCTR5	0000 0000$_H$	32	FFD0 01EC$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Pointer Control Register 6	RSCAN0CFPCTR6	0000 0000$_H$	32	FFD0 01F0$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Pointer Control Register 7	RSCAN0CFPCTR7	0000 0000$_H$	32	FFD0 01F4$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Pointer Control Register 8	RSCAN0CFPCTR8	0000 0000$_H$	32	FFD0 01F8$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Pointer Control Register 9	RSCAN0CFPCTR9	0000 0000$_H$	32	FFD0 01FC$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Pointer Control Register 10	RSCAN0CFPCTR10	0000 0000$_H$	32	FFD0 0200$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Pointer Control Register 11	RSCAN0CFPCTR11	0000 0000$_H$	32	FFD0 0204$_H$	2	8, 16, 32
RSCAN0	FIFO Empty Status Register	RSCAN0FESTS	007F FFFF$_H$	32	FFD0 0238$_H$	2	8, 16, 32
RSCAN0	FIFO Full Status Register	RSCAN0FFSTS	0000 0000$_H$	32	FFD0 023C$_H$	2	8, 16, 32
RSCAN0	FIFO Message Lost Status Register	RSCAN0FMSTS	0000 0000$_H$	32	FFD0 0240$_H$	2	8, 16, 32
RSCAN0	Receive FIFO Buffer Interrupt Flag Status Register	RSCAN0RFISTS	0000 0000$_H$	32	FFD0 0244$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Receive Interrupt Flag Status Register	RSCAN0CFRISTS	0000 0000$_H$	32	FFD0 0248$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Transmit Interrupt Flag Status Register	RSCAN0CFTISTS	0000 0000$_H$	32	FFD0 024C$_H$	2	8, 16, 32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
RSCAN0	Transfer Buffer Control Register 0	RSCAN0TMC0	00_H	8	FFD0 0250_H	2	8
RSCAN0	Transfer Buffer Control Register 1	RSCAN0TMC1	00_H	8	FFD0 0251_H	2	8
RSCAN0	Transfer Buffer Control Register 2	RSCAN0TMC2	00_H	8	FFD0 0252_H	2	8
RSCAN0	Transfer Buffer Control Register 3	RSCAN0TMC3	00_H	8	FFD0 0253_H	2	8
RSCAN0	Transfer Buffer Control Register 4	RSCAN0TMC4	00_H	8	FFD0 0254_H	2	8
RSCAN0	Transfer Buffer Control Register 5	RSCAN0TMC5	00_H	8	FFD0 0255_H	2	8
RSCAN0	Transfer Buffer Control Register 6	RSCAN0TMC6	00_H	8	FFD0 0256_H	2	8
RSCAN0	Transfer Buffer Control Register 7	RSCAN0TMC7	00_H	8	FFD0 0257_H	2	8
RSCAN0	Transfer Buffer Control Register 8	RSCAN0TMC8	00_H	8	FFD0 0258_H	2	8
RSCAN0	Transfer Buffer Control Register 9	RSCAN0TMC9	00_H	8	FFD0 0259_H	2	8
RSCAN0	Transfer Buffer Control Register 10	RSCAN0TMC10	00_H	8	FFD0 $025A_H$	2	8
RSCAN0	Transfer Buffer Control Register 11	RSCAN0TMC11	00_H	8	FFD0 $025B_H$	2	8
RSCAN0	Transfer Buffer Control Register 12	RSCAN0TMC12	00_H	8	FFD0 $025C_H$	2	8
RSCAN0	Transfer Buffer Control Register 13	RSCAN0TMC13	00_H	8	FFD0 $025D_H$	2	8
RSCAN0	Transfer Buffer Control Register 14	RSCAN0TMC14	00_H	8	FFD0 $025E_H$	2	8
RSCAN0	Transfer Buffer Control Register 15	RSCAN0TMC15	00_H	8	FFD0 $025F_H$	2	8
RSCAN0	Transfer Buffer Control Register 16	RSCAN0TMC16	00_H	8	FFD0 0260_H	2	8
RSCAN0	Transfer Buffer Control Register 17	RSCAN0TMC17	00_H	8	FFD0 0261_H	2	8
RSCAN0	Transfer Buffer Control Register 18	RSCAN0TMC18	00_H	8	FFD0 0262_H	2	8
RSCAN0	Transfer Buffer Control Register 19	RSCAN0TMC19	00_H	8	FFD0 0263_H	2	8
RSCAN0	Transfer Buffer Control Register 20	RSCAN0TMC20	00_H	8	FFD0 0264_H	2	8
RSCAN0	Transfer Buffer Control Register 21	RSCAN0TMC21	00_H	8	FFD0 0265_H	2	8
RSCAN0	Transfer Buffer Control Register 22	RSCAN0TMC22	00_H	8	FFD0 0266_H	2	8
RSCAN0	Transfer Buffer Control Register 23	RSCAN0TMC23	00_H	8	FFD0 0267_H	2	8
RSCAN0	Transfer Buffer Control Register 24	RSCAN0TMC24	00_H	8	FFD0 0268_H	2	8
RSCAN0	Transfer Buffer Control Register 25	RSCAN0TMC25	00_H	8	FFD0 0269_H	2	8
RSCAN0	Transfer Buffer Control Register 26	RSCAN0TMC26	00_H	8	FFD0 $026A_H$	2	8
RSCAN0	Transfer Buffer Control Register 27	RSCAN0TMC27	00_H	8	FFD0 $026B_H$	2	8
RSCAN0	Transfer Buffer Control Register 28	RSCAN0TMC28	00_H	8	FFD0 $026C_H$	2	8
RSCAN0	Transfer Buffer Control Register 29	RSCAN0TMC29	00_H	8	FFD0 $026D_H$	2	8
RSCAN0	Transfer Buffer Control Register 30	RSCAN0TMC30	00_H	8	FFD0 $026E_H$	2	8
RSCAN0	Transfer Buffer Control Register 31	RSCAN0TMC31	00_H	8	FFD0 $026F_H$	2	8
RSCAN0	Transfer Buffer Control Register 32	RSCAN0TMC32	00_H	8	FFD0 0270_H	2	8
RSCAN0	Transfer Buffer Control Register 33	RSCAN0TMC33	00_H	8	FFD0 0271_H	2	8
RSCAN0	Transfer Buffer Control Register 34	RSCAN0TMC34	00_H	8	FFD0 0272_H	2	8
RSCAN0	Transfer Buffer Control Register 35	RSCAN0TMC35	00_H	8	FFD0 0273_H	2	8
RSCAN0	Transfer Buffer Control Register 36	RSCAN0TMC36	00_H	8	FFD0 0274_H	2	8
RSCAN0	Transfer Buffer Control Register 37	RSCAN0TMC37	00_H	8	FFD0 0275_H	2	8
RSCAN0	Transfer Buffer Control Register 38	RSCAN0TMC38	00_H	8	FFD0 0276_H	2	8
RSCAN0	Transfer Buffer Control Register 39	RSCAN0TMC39	00_H	8	FFD0 0277_H	2	8
RSCAN0	Transfer Buffer Control Register 40	RSCAN0TMC40	00_H	8	FFD0 0278_H	2	8
RSCAN0	Transfer Buffer Control Register 41	RSCAN0TMC41	00_H	8	FFD0 0279_H	2	8
RSCAN0	Transfer Buffer Control Register 42	RSCAN0TMC42	00_H	8	FFD0 $027A_H$	2	8
RSCAN0	Transfer Buffer Control Register 43	RSCAN0TMC43	00_H	8	FFD0 $027B_H$	2	8
RSCAN0	Transfer Buffer Control Register 44	RSCAN0TMC44	00_H	8	FFD0 $027C_H$	2	8
RSCAN0	Transfer Buffer Control Register 45	RSCAN0TMC45	00_H	8	FFD0 $027D_H$	2	8
RSCAN0	Transfer Buffer Control Register 46	RSCAN0TMC46	00_H	8	FFD0 $027E_H$	2	8

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
RSCAN0	Transfer Buffer Control Register 47	RSCAN0TMC47	00_H	8	FFD0 $027F_H$	2	8
RSCAN0	Transfer Buffer Control Register 48	RSCAN0TMC48	00_H	8	FFD0 0280_H	2	8
RSCAN0	Transfer Buffer Control Register 49	RSCAN0TMC49	00_H	8	FFD0 0281_H	2	8
RSCAN0	Transfer Buffer Control Register 50	RSCAN0TMC50	00_H	8	FFD0 0282_H	2	8
RSCAN0	Transfer Buffer Control Register 51	RSCAN0TMC51	00_H	8	FFD0 0283_H	2	8
RSCAN0	Transfer Buffer Control Register 52	RSCAN0TMC52	00_H	8	FFD0 0284_H	2	8
RSCAN0	Transfer Buffer Control Register 53	RSCAN0TMC53	00_H	8	FFD0 0285_H	2	8
RSCAN0	Transfer Buffer Control Register 54	RSCAN0TMC54	00_H	8	FFD0 0286_H	2	8
RSCAN0	Transfer Buffer Control Register 55	RSCAN0TMC55	00_H	8	FFD0 0287_H	2	8
RSCAN0	Transfer Buffer Control Register 56	RSCAN0TMC56	00_H	8	FFD0 0288_H	2	8
RSCAN0	Transfer Buffer Control Register 57	RSCAN0TMC57	00_H	8	FFD0 0289_H	2	8
RSCAN0	Transfer Buffer Control Register 58	RSCAN0TMC58	00_H	8	FFD0 $028A_H$	2	8
RSCAN0	Transfer Buffer Control Register 59	RSCAN0TMC59	00_H	8	FFD0 $028B_H$	2	8
RSCAN0	Transfer Buffer Control Register 60	RSCAN0TMC60	00_H	8	FFD0 $028C_H$	2	8
RSCAN0	Transfer Buffer Control Register 61	RSCAN0TMC61	00_H	8	FFD0 $028D_H$	2	8
RSCAN0	Transfer Buffer Control Register 62	RSCAN0TMC62	00_H	8	FFD0 $028E_H$	2	8
RSCAN0	Transfer Buffer Control Register 63	RSCAN0TMC63	00_H	8	FFD0 $028F_H$	2	8
RSCAN0	Transfer Buffer Status Register 0	RSCAN0TMSTS0	00_H	8	FFD0 $02D0_H$	2	8
RSCAN0	Transfer Buffer Status Register 1	RSCAN0TMSTS1	00_H	8	FFD0 $02D1_H$	2	8
RSCAN0	Transfer Buffer Status Register 2	RSCAN0TMSTS2	00_H	8	FFD0 $02D2_H$	2	8
RSCAN0	Transfer Buffer Status Register 3	RSCAN0TMSTS3	00_H	8	FFD0 $02D3_H$	2	8
RSCAN0	Transfer Buffer Status Register 4	RSCAN0TMSTS4	00_H	8	FFD0 $02D4_H$	2	8
RSCAN0	Transfer Buffer Status Register 5	RSCAN0TMSTS5	00_H	8	FFD0 $02D5_H$	2	8
RSCAN0	Transfer Buffer Status Register 6	RSCAN0TMSTS6	00_H	8	FFD0 $02D6_H$	2	8
RSCAN0	Transfer Buffer Status Register 7	RSCAN0TMSTS7	00_H	8	FFD0 $02D7_H$	2	8
RSCAN0	Transfer Buffer Status Register 8	RSCAN0TMSTS8	00_H	8	FFD0 $02D8_H$	2	8
RSCAN0	Transfer Buffer Status Register 9	RSCAN0TMSTS9	00_H	8	FFD0 $02D9_H$	2	8
RSCAN0	Transfer Buffer Status Register 10	RSCAN0TMSTS10	00_H	8	FFD0 $02DA_H$	2	8
RSCAN0	Transfer Buffer Status Register 11	RSCAN0TMSTS11	00_H	8	FFD0 $02DB_H$	2	8
RSCAN0	Transfer Buffer Status Register 12	RSCAN0TMSTS12	00_H	8	FFD0 $02DC_H$	2	8
RSCAN0	Transfer Buffer Status Register 13	RSCAN0TMSTS13	00_H	8	FFD0 $02DD_H$	2	8
RSCAN0	Transfer Buffer Status Register 14	RSCAN0TMSTS14	00_H	8	FFD0 $02DE_H$	2	8
RSCAN0	Transfer Buffer Status Register 15	RSCAN0TMSTS15	00_H	8	FFD0 $02DF_H$	2	8
RSCAN0	Transfer Buffer Status Register 16	RSCAN0TMSTS16	00_H	8	FFD0 $02E0_H$	2	8
RSCAN0	Transfer Buffer Status Register 17	RSCAN0TMSTS17	00_H	8	FFD0 $02E1_H$	2	8
RSCAN0	Transfer Buffer Status Register 18	RSCAN0TMSTS18	00_H	8	FFD0 $02E2_H$	2	8
RSCAN0	Transfer Buffer Status Register 19	RSCAN0TMSTS19	00_H	8	FFD0 $02E3_H$	2	8
RSCAN0	Transfer Buffer Status Register 20	RSCAN0TMSTS20	00_H	8	FFD0 $02E4_H$	2	8
RSCAN0	Transfer Buffer Status Register 21	RSCAN0TMSTS21	00_H	8	FFD0 $02E5_H$	2	8
RSCAN0	Transfer Buffer Status Register 22	RSCAN0TMSTS22	00_H	8	FFD0 $02E6_H$	2	8
RSCAN0	Transfer Buffer Status Register 23	RSCAN0TMSTS23	00_H	8	FFD0 $02E7_H$	2	8
RSCAN0	Transfer Buffer Status Register 24	RSCAN0TMSTS24	00_H	8	FFD0 $02E8_H$	2	8
RSCAN0	Transfer Buffer Status Register 25	RSCAN0TMSTS25	00_H	8	FFD0 $02E9_H$	2	8
RSCAN0	Transfer Buffer Status Register 26	RSCAN0TMSTS26	00_H	8	FFD0 $02EA_H$	2	8
RSCAN0	Transfer Buffer Status Register 27	RSCAN0TMSTS27	00_H	8	FFD0 $02EB_H$	2	8
RSCAN0	Transfer Buffer Status Register 28	RSCAN0TMSTS28	00_H	8	FFD0 $02EC_H$	2	8
RSCAN0	Transfer Buffer Status Register 29	RSCAN0TMSTS29	00_H	8	FFD0 $02ED_H$	2	8

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
RSCAN0	Transfer Buffer Status Register 30	RSCAN0TMSTS30	00_H	8	FFD0 02EE$_H$	2	8
RSCAN0	Transfer Buffer Status Register 31	RSCAN0TMSTS31	00_H	8	FFD0 02EF$_H$	2	8
RSCAN0	Transfer Buffer Status Register 32	RSCAN0TMSTS32	00_H	8	FFD0 02F0$_H$	2	8
RSCAN0	Transfer Buffer Status Register 33	RSCAN0TMSTS33	00_H	8	FFD0 02F1$_H$	2	8
RSCAN0	Transfer Buffer Status Register 34	RSCAN0TMSTS34	00_H	8	FFD0 02F2$_H$	2	8
RSCAN0	Transfer Buffer Status Register 35	RSCAN0TMSTS35	00_H	8	FFD0 02F3$_H$	2	8
RSCAN0	Transfer Buffer Status Register 36	RSCAN0TMSTS36	00_H	8	FFD0 02F4$_H$	2	8
RSCAN0	Transfer Buffer Status Register 37	RSCAN0TMSTS37	00_H	8	FFD0 02F5$_H$	2	8
RSCAN0	Transfer Buffer Status Register 38	RSCAN0TMSTS38	00_H	8	FFD0 02F6$_H$	2	8
RSCAN0	Transfer Buffer Status Register 39	RSCAN0TMSTS39	00_H	8	FFD0 02F7$_H$	2	8
RSCAN0	Transfer Buffer Status Register 40	RSCAN0TMSTS40	00_H	8	FFD0 02F8$_H$	2	8
RSCAN0	Transfer Buffer Status Register 41	RSCAN0TMSTS41	00_H	8	FFD0 02F9$_H$	2	8
RSCAN0	Transfer Buffer Status Register 42	RSCAN0TMSTS42	00_H	8	FFD0 02FA$_H$	2	8
RSCAN0	Transfer Buffer Status Register 43	RSCAN0TMSTS43	00_H	8	FFD0 02FB$_H$	2	8
RSCAN0	Transfer Buffer Status Register 44	RSCAN0TMSTS44	00_H	8	FFD0 02FC$_H$	2	8
RSCAN0	Transfer Buffer Status Register 45	RSCAN0TMSTS45	00_H	8	FFD0 02FD$_H$	2	8
RSCAN0	Transfer Buffer Status Register 46	RSCAN0TMSTS46	00_H	8	FFD0 02FE$_H$	2	8
RSCAN0	Transfer Buffer Status Register 47	RSCAN0TMSTS47	00_H	8	FFD0 02FF$_H$	2	8
RSCAN0	Transfer Buffer Status Register 48	RSCAN0TMSTS48	00_H	8	FFD0 0300$_H$	2	8
RSCAN0	Transfer Buffer Status Register 49	RSCAN0TMSTS49	00_H	8	FFD0 0301$_H$	2	8
RSCAN0	Transfer Buffer Status Register 50	RSCAN0TMSTS50	00_H	8	FFD0 0302$_H$	2	8
RSCAN0	Transfer Buffer Status Register 51	RSCAN0TMSTS51	00_H	8	FFD0 0303$_H$	2	8
RSCAN0	Transfer Buffer Status Register 52	RSCAN0TMSTS52	00_H	8	FFD0 0304$_H$	2	8
RSCAN0	Transfer Buffer Status Register 53	RSCAN0TMSTS53	00_H	8	FFD0 0305$_H$	2	8
RSCAN0	Transfer Buffer Status Register 54	RSCAN0TMSTS54	00_H	8	FFD0 0306$_H$	2	8
RSCAN0	Transfer Buffer Status Register 55	RSCAN0TMSTS55	00_H	8	FFD0 0307$_H$	2	8
RSCAN0	Transfer Buffer Status Register 56	RSCAN0TMSTS56	00_H	8	FFD0 0308$_H$	2	8
RSCAN0	Transfer Buffer Status Register 57	RSCAN0TMSTS57	00_H	8	FFD0 0309$_H$	2	8
RSCAN0	Transfer Buffer Status Register 58	RSCAN0TMSTS58	00_H	8	FFD0 030A$_H$	2	8
RSCAN0	Transfer Buffer Status Register 59	RSCAN0TMSTS59	00_H	8	FFD0 030B$_H$	2	8
RSCAN0	Transfer Buffer Status Register 60	RSCAN0TMSTS60	00_H	8	FFD0 030C$_H$	2	8
RSCAN0	Transfer Buffer Status Register 61	RSCAN0TMSTS61	00_H	8	FFD0 030D$_H$	2	8
RSCAN0	Transfer Buffer Status Register 62	RSCAN0TMSTS62	00_H	8	FFD0 030E$_H$	2	8
RSCAN0	Transfer Buffer Status Register 63	RSCAN0TMSTS63	00_H	8	FFD0 030F$_H$	2	8
RSCAN0	Transfer Buffer Transmission Request Status Register 0	RSCAN0TMTRSTS0	0000 0000_H	32	FFD0 0350$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Transmission Request Status Register 1	RSCAN0TMTRSTS1	0000 0000_H	32	FFD0 0354$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Transmission Abort Request Status Register 0	RSCAN0TMTARSTS0	0000 0000_H	32	FFD0 0360$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Transmission Abort Request Status Register 1	RSCAN0TMTARSTS1	0000 0000_H	32	FFD0 0364$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Transmission Complete Status Register 0	RSCAN0TMTCSTS0	0000 0000_H	32	FFD0 0370$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Transmission Complete Status Register 1	RSCAN0TMTCSTS1	0000 0000_H	32	FFD0 0374$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Transmission Abort Status Register 0	RSCAN0TMTASTS0	0000 0000_H	32	FFD0 0380$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Interrupt Enable Configuration Register 0	RSCAN0TMTASTS1	0000 0000_H	32	FFD0 0384$_H$	2	8, 16, 32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
RSCAN0	Transfer Buffer Interrupt Enable Configuration Register 1	RSCAN0TMIEC0	0000 0000$_H$	32	FFD0 0390$_H$	2	8, 16, 32
RSCAN0	Transmission Queue Configuration and Control Register 0	RSCAN0TMIEC1	0000 0000$_H$	32	FFD0 0394$_H$	2	8, 16, 32
RSCAN0	Transmission Queue Configuration and Control Register 1	RSCAN0TXQCC0	0000 0000$_H$	32	FFD0 03A0$_H$	2	8, 16, 32
RSCAN0	Transmission Queue Configuration and Control Register 2	RSCAN0TXQCC1	0000 0000$_H$	32	FFD0 03A4$_H$	2	8, 16, 32
RSCAN0	Transmission Queue Configuration and Control Register 3	RSCAN0TXQCC2	0000 0000$_H$	32	FFD0 03A8$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Interrupt Enable Configuration Register 0	RSCAN0TXQCC3	0000 0000$_H$	32	FFD0 03AC$_H$	2	8, 16, 32
RSCAN0	Transmission Queue Status Register 0	RSCAN0TXQSTS0	0000 0001$_H$	32	FFD0 03C0$_H$	2	8, 16, 32
RSCAN0	Transmission Queue Status Register 1	RSCAN0TXQSTS1	0000 0001$_H$	32	FFD0 03C4$_H$	2	8, 16, 32
RSCAN0	Transmission Queue Status Register 2	RSCAN0TXQSTS2	0000 0001$_H$	32	FFD0 03C8$_H$	2	8, 16, 32
RSCAN0	Transmission Queue Status Register 3	RSCAN0TXQSTS3	0000 0001$_H$	32	FFD0 03CC$_H$	2	8, 16, 32
RSCAN0	Transmission Queue Pointer Control Register 0	RSCAN0TXQPCTR0	0000 0000$_H$	32	FFD0 03E0$_H$	2	8, 16, 32
RSCAN0	Transmission Queue Pointer Control Register 1	RSCAN0TXQPCTR1	0000 0000$_H$	32	FFD0 03E4$_H$	2	8, 16, 32
RSCAN0	Transmission Queue Pointer Control Register 2	RSCAN0TXQPCTR2	0000 0000$_H$	32	FFD0 03E8$_H$	2	8, 16, 32
RSCAN0	Transmission Queue Pointer Control Register 3	RSCAN0TXQPCTR3	0000 0000$_H$	32	FFD0 03EC$_H$	2	8, 16, 32
RSCAN0	Transmission History Configuration and Control Register 0	RSCAN0THLCC0	0000 0000$_H$	32	FFD0 0400$_H$	2	8, 16, 32
RSCAN0	Transmission History Configuration and Control Register 1	RSCAN0THLCC1	0000 0000$_H$	32	FFD0 0404$_H$	2	8, 16, 32
RSCAN0	Transmission History Configuration and Control Register 2	RSCAN0THLCC2	0000 0000$_H$	32	FFD0 0408$_H$	2	8, 16, 32
RSCAN0	Transmission History Configuration and Control Register 3	RSCAN0THLCC3	0000 0000$_H$	32	FFD0 040C$_H$	2	8, 16, 32
RSCAN0	Transmission History Status Register 0	RSCAN0THLSTS0	0000 0001$_H$	32	FFD0 0420$_H$	2	8, 16, 32
RSCAN0	Transmission History Status Register 1	RSCAN0THLSTS1	0000 0001$_H$	32	FFD0 0424$_H$	2	8, 16, 32
RSCAN0	Transmission History Status Register 2	RSCAN0THLSTS2	0000 0001$_H$	32	FFD0 0428$_H$	2	8, 16, 32
RSCAN0	Transmission History Status Register 3	RSCAN0THLSTS3	0000 0001$_H$	32	FFD0 042C$_H$	2	8, 16, 32
RSCAN0	Transmission History Pointer Control Register 0	RSCAN0THLPCTR0	0000 0000$_H$	32	FFD0 0440$_H$	2	8, 16, 32
RSCAN0	Transmission History Pointer Control Register 1	RSCAN0THLPCTR1	0000 0000$_H$	32	FFD0 0444$_H$	2	8, 16, 32
RSCAN0	Transmission History Pointer Control Register 2	RSCAN0THLPCTR2	0000 0000$_H$	32	FFD0 0448$_H$	2	8, 16, 32
RSCAN0	Transmission History Pointer Control Register 3	RSCAN0THLPCTR3	0000 0000$_H$	32	FFD0 044C$_H$	2	8, 16, 32
RSCAN0	Global TX Interrupt Status Register 0	RSCAN0GTINTSTS0	0000 0000$_H$	32	FFD0 0460$_H$	2	8, 16, 32
RSCAN0	Global Test Configuration Register	RSCAN0GTSTCFG	0000 0000$_H$	32	FFD0 0468$_H$	2	8, 16, 32
RSCAN0	Global Test Control Register	RSCAN0GTSTCTR	0000 0000$_H$	32	FFD0 046C$_H$	2	8, 16, 32
RSCAN0	Global Lock Key Register	RSCAN0GLOCKK	0000 0000$_H$	32	FFD0 047C$_H$	2	16, 32
RSCAN0	Receive Rule ID Register 0	RSCAN0GAFLID0	0000 0000$_H$	32	FFD0 0500$_H$	2	8, 16, 32
RSCAN0	Receive Rule Mask Register 0	RSCAN0GAFLM0	0000 0000$_H$	32	FFD0 0504$_H$	2	8, 16, 32
RSCAN0	Receive Rule Pointer 0 Register 0	RSCAN0GAFLP00	0000 0000$_H$	32	FFD0 0508$_H$	2	8, 16, 32
RSCAN0	Receive Rule Pointer 1 Register 0	RSCAN0GAFLP10	0000 0000$_H$	32	FFD0 050C$_H$	2	8, 16, 32
RSCAN0	Receive Rule ID Register 1	RSCAN0GAFLID1	0000 0000$_H$	32	FFD0 0510$_H$	2	8, 16, 32
RSCAN0	Receive Rule Mask Register 1	RSCAN0GAFLM1	0000 0000$_H$	32	FFD0 0514$_H$	2	8, 16, 32

(46/201)

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
RSCAN0	Receive Rule Pointer 0 Register 1	RSCAN0GAFLP01	0000 0000$_H$	32	FFD0 0518$_H$	2	8, 16, 32
RSCAN0	Receive Rule Pointer 1 Register 1	RSCAN0GAFLP11	0000 0000$_H$	32	FFD0 051C$_H$	2	8, 16, 32
RSCAN0	Receive Rule ID Register 2	RSCAN0GAFLID2	0000 0000$_H$	32	FFD0 0520$_H$	2	8, 16, 32
RSCAN0	Receive Rule Mask Register 2	RSCAN0GAFLM2	0000 0000$_H$	32	FFD0 0524$_H$	2	8, 16, 32
RSCAN0	Receive Rule Pointer 0 Register 2	RSCAN0GAFLP02	0000 0000$_H$	32	FFD0 0528$_H$	2	8, 16, 32
RSCAN0	Receive Rule Pointer 1 Register 2	RSCAN0GAFLP12	0000 0000$_H$	32	FFD0 052C$_H$	2	8, 16, 32
RSCAN0	Receive Rule ID Register 3	RSCAN0GAFLID3	0000 0000$_H$	32	FFD0 0530$_H$	2	8, 16, 32
RSCAN0	Receive Rule Mask Register 3	RSCAN0GAFLM3	0000 0000$_H$	32	FFD0 0534$_H$	2	8, 16, 32
RSCAN0	Receive Rule Pointer 0 Register 3	RSCAN0GAFLP03	0000 0000$_H$	32	FFD0 0538$_H$	2	8, 16, 32
RSCAN0	Receive Rule Pointer 1 Register 3	RSCAN0GAFLP13	0000 0000$_H$	32	FFD0 053C$_H$	2	8, 16, 32
RSCAN0	Receive Rule ID Register 4	RSCAN0GAFLID4	0000 0000$_H$	32	FFD0 0540$_H$	2	8, 16, 32
RSCAN0	Receive Rule Mask Register 4	RSCAN0GAFLM4	0000 0000$_H$	32	FFD0 0544$_H$	2	8, 16, 32
RSCAN0	Receive Rule Pointer 0 Register 4	RSCAN0GAFLP04	0000 0000$_H$	32	FFD0 0548$_H$	2	8, 16, 32
RSCAN0	Receive Rule Pointer 1 Register 4	RSCAN0GAFLP14	0000 0000$_H$	32	FFD0 054C$_H$	2	8, 16, 32
RSCAN0	Receive Rule ID Register 5	RSCAN0GAFLID5	0000 0000$_H$	32	FFD0 0550$_H$	2	8, 16, 32
RSCAN0	Receive Rule Mask Register 5	RSCAN0GAFLM5	0000 0000$_H$	32	FFD0 0554$_H$	2	8, 16, 32
RSCAN0	Receive Rule Pointer 0 Register 5	RSCAN0GAFLP05	0000 0000$_H$	32	FFD0 0558$_H$	2	8, 16, 32
RSCAN0	Receive Rule Pointer 1 Register 5	RSCAN0GAFLP15	0000 0000$_H$	32	FFD0 055C$_H$	2	8, 16, 32
RSCAN0	Receive Rule ID Register 6	RSCAN0GAFLID6	0000 0000$_H$	32	FFD0 0560$_H$	2	8, 16, 32
RSCAN0	Receive Rule Mask Register 6	RSCAN0GAFLM6	0000 0000$_H$	32	FFD0 0564$_H$	2	8, 16, 32
RSCAN0	Receive Rule Pointer 0 Register 6	RSCAN0GAFLP06	0000 0000$_H$	32	FFD0 0568$_H$	2	8, 16, 32
RSCAN0	Receive Rule Pointer 1 Register 6	RSCAN0GAFLP16	0000 0000$_H$	32	FFD0 056C$_H$	2	8, 16, 32
RSCAN0	Receive Rule ID Register 7	RSCAN0GAFLID7	0000 0000$_H$	32	FFD0 0570$_H$	2	8, 16, 32
RSCAN0	Receive Rule Mask Register 7	RSCAN0GAFLM7	0000 0000$_H$	32	FFD0 0574$_H$	2	8, 16, 32
RSCAN0	Receive Rule Pointer 0 Register 7	RSCAN0GAFLP07	0000 0000$_H$	32	FFD0 0578$_H$	2	8, 16, 32
RSCAN0	Receive Rule Pointer 1 Register 7	RSCAN0GAFLP17	0000 0000$_H$	32	FFD0 057C$_H$	2	8, 16, 32
RSCAN0	Receive Rule ID Register 8	RSCAN0GAFLID8	0000 0000$_H$	32	FFD0 0580$_H$	2	8, 16, 32
RSCAN0	Receive Rule Mask Register 8	RSCAN0GAFLM8	0000 0000$_H$	32	FFD0 0584$_H$	2	8, 16, 32
RSCAN0	Receive Rule Pointer 0 Register 8	RSCAN0GAFLP08	0000 0000$_H$	32	FFD0 0588$_H$	2	8, 16, 32
RSCAN0	Receive Rule Pointer 1 Register 8	RSCAN0GAFLP18	0000 0000$_H$	32	FFD0 058C$_H$	2	8, 16, 32
RSCAN0	Receive Rule ID Register 9	RSCAN0GAFLID9	0000 0000$_H$	32	FFD0 0590$_H$	2	8, 16, 32
RSCAN0	Receive Rule Mask Register 9	RSCAN0GAFLM9	0000 0000$_H$	32	FFD0 0594$_H$	2	8, 16, 32
RSCAN0	Receive Rule Pointer 0 Register 9	RSCAN0GAFLP09	0000 0000$_H$	32	FFD0 0598$_H$	2	8, 16, 32
RSCAN0	Receive Rule Pointer 1 Register 9	RSCAN0GAFLP19	0000 0000$_H$	32	FFD0 059C$_H$	2	8, 16, 32
RSCAN0	Receive Rule ID Register 10	RSCAN0GAFLID10	0000 0000$_H$	32	FFD0 05A0$_H$	2	8, 16, 32
RSCAN0	Receive Rule Mask Register 10	RSCAN0GAFLM10	0000 0000$_H$	32	FFD0 05A4$_H$	2	8, 16, 32
RSCAN0	Receive Rule Pointer 0 Register 10	RSCAN0GAFLP010	0000 0000$_H$	32	FFD0 05A8$_H$	2	8, 16, 32
RSCAN0	Receive Rule Pointer 1 Register 10	RSCAN0GAFLP110	0000 0000$_H$	32	FFD0 05AC$_H$	2	8, 16, 32
RSCAN0	Receive Rule ID Register 11	RSCAN0GAFLID11	0000 0000$_H$	32	FFD0 05B0$_H$	2	8, 16, 32
RSCAN0	Receive Rule Mask Register 11	RSCAN0GAFLM11	0000 0000$_H$	32	FFD0 05B4$_H$	2	8, 16, 32
RSCAN0	Receive Rule Pointer 0 Register 11	RSCAN0GAFLP011	0000 0000$_H$	32	FFD0 05B8$_H$	2	8, 16, 32
RSCAN0	Receive Rule Pointer 1 Register 11	RSCAN0GAFLP111	0000 0000$_H$	32	FFD0 05BC$_H$	2	8, 16, 32
RSCAN0	Receive Rule ID Register 12	RSCAN0GAFLID12	0000 0000$_H$	32	FFD0 05C0$_H$	2	8, 16, 32
RSCAN0	Receive Rule Mask Register 12	RSCAN0GAFLM12	0000 0000$_H$	32	FFD0 05C4$_H$	2	8, 16, 32
RSCAN0	Receive Rule Pointer 0 Register 12	RSCAN0GAFLP012	0000 0000$_H$	32	FFD0 05C8$_H$	2	8, 16, 32
RSCAN0	Receive Rule Pointer 1 Register 12	RSCAN0GAFLP112	0000 0000$_H$	32	FFD0 05CC$_H$	2	8, 16, 32
RSCAN0	Receive Rule ID Register 13	RSCAN0GAFLID13	0000 0000$_H$	32	FFD0 05D0$_H$	2	8, 16, 32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
RSCAN0	Receive Rule Mask Register 13	RSCAN0GAFLM13	0000 0000$_H$	32	FFD0 05D4$_H$	2	8, 16, 32
RSCAN0	Receive Rule Pointer 0 Register 13	RSCAN0GAFLP013	0000 0000$_H$	32	FFD0 05D8$_H$	2	8, 16, 32
RSCAN0	Receive Rule Pointer 1 Register 13	RSCAN0GAFLP113	0000 0000$_H$	32	FFD0 05DC$_H$	2	8, 16, 32
RSCAN0	Receive Rule ID Register 14	RSCAN0GAFLID14	0000 0000$_H$	32	FFD0 05E0$_H$	2	8, 16, 32
RSCAN0	Receive Rule Mask Register 14	RSCAN0GAFLM14	0000 0000$_H$	32	FFD0 05E4$_H$	2	8, 16, 32
RSCAN0	Receive Rule Pointer 0 Register 14	RSCAN0GAFLP014	0000 0000$_H$	32	FFD0 05E8$_H$	2	8, 16, 32
RSCAN0	Receive Rule Pointer 1 Register 14	RSCAN0GAFLP114	0000 0000$_H$	32	FFD0 05EC$_H$	2	8, 16, 32
RSCAN0	Receive Rule ID Register 15	RSCAN0GAFLID15	0000 0000$_H$	32	FFD0 05F0$_H$	2	8, 16, 32
RSCAN0	Receive Rule Mask Register 15	RSCAN0GAFLM15	0000 0000$_H$	32	FFD0 05F4$_H$	2	8, 16, 32
RSCAN0	Receive Rule Pointer 0 Register 15	RSCAN0GAFLP015	0000 0000$_H$	32	FFD0 05F8$_H$	2	8, 16, 32
RSCAN0	Receive Rule Pointer 1 Register 15	RSCAN0GAFLP115	0000 0000$_H$	32	FFD0 05FC$_H$	2	8, 16, 32
RSCAN0	Receive Buffer ID Register 0	RSCAN0RMID0	0000 0000$_H$	32	FFD0 0600$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 0	RSCAN0RMPTR0	0000 0000$_H$	32	FFD0 0604$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 0	RSCAN0RMDF00	0000 0000$_H$	32	FFD0 0608$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 0	RSCAN0RMDF10	0000 0000$_H$	32	FFD0 060C$_H$	2	8, 16, 32
RSCAN0	Receive Buffer ID Register 1	RSCAN0RMID1	0000 0000$_H$	32	FFD0 0610$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 1	RSCAN0RMPTR1	0000 0000$_H$	32	FFD0 0614$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 1	RSCAN0RMDF01	0000 0000$_H$	32	FFD0 0618$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 1	RSCAN0RMDF11	0000 0000$_H$	32	FFD0 061C$_H$	2	8, 16, 32
RSCAN0	Receive Buffer ID Register 2	RSCAN0RMID2	0000 0000$_H$	32	FFD0 0620$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 2	RSCAN0RMPTR2	0000 0000$_H$	32	FFD0 0624$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 2	RSCAN0RMDF02	0000 0000$_H$	32	FFD0 0628$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 2	RSCAN0RMDF12	0000 0000$_H$	32	FFD0 062C$_H$	2	8, 16, 32
RSCAN0	Receive Buffer ID Register 3	RSCAN0RMID3	0000 0000$_H$	32	FFD0 0630$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 3	RSCAN0RMPTR3	0000 0000$_H$	32	FFD0 0634$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 3	RSCAN0RMDF03	0000 0000$_H$	32	FFD0 0638$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 3	RSCAN0RMDF13	0000 0000$_H$	32	FFD0 063C$_H$	2	8, 16, 32
RSCAN0	Receive Buffer ID Register 4	RSCAN0RMID4	0000 0000$_H$	32	FFD0 0640$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 4	RSCAN0RMPTR4	0000 0000$_H$	32	FFD0 0644$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 4	RSCAN0RMDF04	0000 0000$_H$	32	FFD0 0648$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 4	RSCAN0RMDF14	0000 0000$_H$	32	FFD0 064C$_H$	2	8, 16, 32
RSCAN0	Receive Buffer ID Register 5	RSCAN0RMID5	0000 0000$_H$	32	FFD0 0650$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 5	RSCAN0RMPTR5	0000 0000$_H$	32	FFD0 0654$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 5	RSCAN0RMDF05	0000 0000$_H$	32	FFD0 0658$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 5	RSCAN0RMDF15	0000 0000$_H$	32	FFD0 065C$_H$	2	8, 16, 32
RSCAN0	Receive Buffer ID Register 6	RSCAN0RMID6	0000 0000$_H$	32	FFD0 0660$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 6	RSCAN0RMPTR6	0000 0000$_H$	32	FFD0 0664$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 6	RSCAN0RMDF06	0000 0000$_H$	32	FFD0 0668$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 6	RSCAN0RMDF16	0000 0000$_H$	32	FFD0 066C$_H$	2	8, 16, 32
RSCAN0	Receive Buffer ID Register 7	RSCAN0RMID7	0000 0000$_H$	32	FFD0 0670$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 7	RSCAN0RMPTR7	0000 0000$_H$	32	FFD0 0674$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 7	RSCAN0RMDF07	0000 0000$_H$	32	FFD0 0678$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 7	RSCAN0RMDF17	0000 0000$_H$	32	FFD0 067C$_H$	2	8, 16, 32
RSCAN0	Receive Buffer ID Register 8	RSCAN0RMID8	0000 0000$_H$	32	FFD0 0680$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 8	RSCAN0RMPTR8	0000 0000$_H$	32	FFD0 0684$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 8	RSCAN0RMDF08	0000 0000$_H$	32	FFD0 0688$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 8	RSCAN0RMDF18	0000 0000$_H$	32	FFD0 068C$_H$	2	8, 16, 32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
RSCAN0	Receive Buffer ID Register 9	RSCAN0RMID9	0000 0000$_H$	32	FFD0 0690$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 9	RSCAN0RMPTR9	0000 0000$_H$	32	FFD0 0694$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 9	RSCAN0RMDF09	0000 0000$_H$	32	FFD0 0698$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 9	RSCAN0RMDF19	0000 0000$_H$	32	FFD0 069C$_H$	2	8, 16, 32
RSCAN0	Receive Buffer ID Register 10	RSCAN0RMID10	0000 0000$_H$	32	FFD0 06A0$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 10	RSCAN0RMPTR10	0000 0000$_H$	32	FFD0 06A4$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 10	RSCAN0RMDF010	0000 0000$_H$	32	FFD0 06A8$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 10	RSCAN0RMDF110	0000 0000$_H$	32	FFD0 06AC$_H$	2	8, 16, 32
RSCAN0	Receive Buffer ID Register 11	RSCAN0RMID11	0000 0000$_H$	32	FFD0 06B0$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 11	RSCAN0RMPTR11	0000 0000$_H$	32	FFD0 06B4$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 11	RSCAN0RMDF011	0000 0000$_H$	32	FFD0 06B8$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 11	RSCAN0RMDF111	0000 0000$_H$	32	FFD0 06BC$_H$	2	8, 16, 32
RSCAN0	Receive Buffer ID Register 12	RSCAN0RMID12	0000 0000$_H$	32	FFD0 06C0$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 12	RSCAN0RMPTR12	0000 0000$_H$	32	FFD0 06C4$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 12	RSCAN0RMDF012	0000 0000$_H$	32	FFD0 06C8$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 12	RSCAN0RMDF112	0000 0000$_H$	32	FFD0 06CC$_H$	2	8, 16, 32
RSCAN0	Receive Buffer ID Register 13	RSCAN0RMID13	0000 0000$_H$	32	FFD0 06D0$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 13	RSCAN0RMPTR13	0000 0000$_H$	32	FFD0 06D4$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 13	RSCAN0RMDF013	0000 0000$_H$	32	FFD0 06D8$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 13	RSCAN0RMDF113	0000 0000$_H$	32	FFD0 06DC$_H$	2	8, 16, 32
RSCAN0	Receive Buffer ID Register 14	RSCAN0RMID14	0000 0000$_H$	32	FFD0 06E0$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 14	RSCAN0RMPTR14	0000 0000$_H$	32	FFD0 06E4$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 14	RSCAN0RMDF014	0000 0000$_H$	32	FFD0 06E8$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 14	RSCAN0RMDF114	0000 0000$_H$	32	FFD0 06EC$_H$	2	8, 16, 32
RSCAN0	Receive Buffer ID Register 15	RSCAN0RMID15	0000 0000$_H$	32	FFD0 06F0$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 15	RSCAN0RMPTR15	0000 0000$_H$	32	FFD0 06F4$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 15	RSCAN0RMDF015	0000 0000$_H$	32	FFD0 06F8$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 15	RSCAN0RMDF115	0000 0000$_H$	32	FFD0 06FC$_H$	2	8, 16, 32
RSCAN0	Receive Buffer ID Register 16	RSCAN0RMID16	0000 0000$_H$	32	FFD0 0700$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 16	RSCAN0RMPTR16	0000 0000$_H$	32	FFD0 0704$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 16	RSCAN0RMDF016	0000 0000$_H$	32	FFD0 0708$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 16	RSCAN0RMDF116	0000 0000$_H$	32	FFD0 070C$_H$	2	8, 16, 32
RSCAN0	Receive Buffer ID Register 17	RSCAN0RMID17	0000 0000$_H$	32	FFD0 0710$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 17	RSCAN0RMPTR17	0000 0000$_H$	32	FFD0 0714$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 17	RSCAN0RMDF017	0000 0000$_H$	32	FFD0 0718$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 17	RSCAN0RMDF117	0000 0000$_H$	32	FFD0 071C$_H$	2	8, 16, 32
RSCAN0	Receive Buffer ID Register 18	RSCAN0RMID18	0000 0000$_H$	32	FFD0 0720$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 18	RSCAN0RMPTR18	0000 0000$_H$	32	FFD0 0724$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 18	RSCAN0RMDF018	0000 0000$_H$	32	FFD0 0728$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 18	RSCAN0RMDF118	0000 0000$_H$	32	FFD0 072C$_H$	2	8, 16, 32
RSCAN0	Receive Buffer ID Register 19	RSCAN0RMID19	0000 0000$_H$	32	FFD0 0730$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 19	RSCAN0RMPTR19	0000 0000$_H$	32	FFD0 0734$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 19	RSCAN0RMDF019	0000 0000$_H$	32	FFD0 0738$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 19	RSCAN0RMDF119	0000 0000$_H$	32	FFD0 073C$_H$	2	8, 16, 32
RSCAN0	Receive Buffer ID Register 20	RSCAN0RMID20	0000 0000$_H$	32	FFD0 0740$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 20	RSCAN0RMPTR20	0000 0000$_H$	32	FFD0 0744$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 20	RSCAN0RMDF020	0000 0000$_H$	32	FFD0 0748$_H$	2	8, 16, 32

(49/201)

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
RSCAN0	Receive Buffer Data Field 1 Register 20	RSCAN0RMDF120	0000 0000$_H$	32	FFD0 074C$_H$	2	8, 16, 32
RSCAN0	Receive Buffer ID Register 21	RSCAN0RMID21	0000 0000$_H$	32	FFD0 0750$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 21	RSCAN0RMPTR21	0000 0000$_H$	32	FFD0 0754$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 21	RSCAN0RMDF021	0000 0000$_H$	32	FFD0 0758$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 21	RSCAN0RMDF121	0000 0000$_H$	32	FFD0 075C$_H$	2	8, 16, 32
RSCAN0	Receive Buffer ID Register 22	RSCAN0RMID22	0000 0000$_H$	32	FFD0 0760$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 22	RSCAN0RMPTR22	0000 0000$_H$	32	FFD0 0764$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 22	RSCAN0RMDF022	0000 0000$_H$	32	FFD0 0768$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 22	RSCAN0RMDF122	0000 0000$_H$	32	FFD0 076C$_H$	2	8, 16, 32
RSCAN0	Receive Buffer ID Register 23	RSCAN0RMID23	0000 0000$_H$	32	FFD0 0770$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 23	RSCAN0RMPTR23	0000 0000$_H$	32	FFD0 0774$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 23	RSCAN0RMDF023	0000 0000$_H$	32	FFD0 0778$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 23	RSCAN0RMDF123	0000 0000$_H$	32	FFD0 077C$_H$	2	8, 16, 32
RSCAN0	Receive Buffer ID Register 24	RSCAN0RMID24	0000 0000$_H$	32	FFD0 0780$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 24	RSCAN0RMPTR24	0000 0000$_H$	32	FFD0 0784$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 24	RSCAN0RMDF024	0000 0000$_H$	32	FFD0 0788$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 24	RSCAN0RMDF124	0000 0000$_H$	32	FFD0 078C$_H$	2	8, 16, 32
RSCAN0	Receive Buffer ID Register 25	RSCAN0RMID25	0000 0000$_H$	32	FFD0 0790$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 25	RSCAN0RMPTR25	0000 0000$_H$	32	FFD0 0794$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 25	RSCAN0RMDF025	0000 0000$_H$	32	FFD0 0798$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 25	RSCAN0RMDF125	0000 0000$_H$	32	FFD0 079C$_H$	2	8, 16, 32
RSCAN0	Receive Buffer ID Register 26	RSCAN0RMID26	0000 0000$_H$	32	FFD0 07A0$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 26	RSCAN0RMPTR26	0000 0000$_H$	32	FFD0 07A4$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 26	RSCAN0RMDF026	0000 0000$_H$	32	FFD0 07A8$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 26	RSCAN0RMDF126	0000 0000$_H$	32	FFD0 07AC$_H$	2	8, 16, 32
RSCAN0	Receive Buffer ID Register 27	RSCAN0RMID27	0000 0000$_H$	32	FFD0 07B0$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 27	RSCAN0RMPTR27	0000 0000$_H$	32	FFD0 07B4$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 27	RSCAN0RMDF027	0000 0000$_H$	32	FFD0 07B8$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 27	RSCAN0RMDF127	0000 0000$_H$	32	FFD0 07BC$_H$	2	8, 16, 32
RSCAN0	Receive Buffer ID Register 28	RSCAN0RMID28	0000 0000$_H$	32	FFD0 07C0$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 28	RSCAN0RMPTR28	0000 0000$_H$	32	FFD0 07C4$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 28	RSCAN0RMDF028	0000 0000$_H$	32	FFD0 07C8$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 28	RSCAN0RMDF128	0000 0000$_H$	32	FFD0 07CC$_H$	2	8, 16, 32
RSCAN0	Receive Buffer ID Register 29	RSCAN0RMID29	0000 0000$_H$	32	FFD0 07D0$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 29	RSCAN0RMPTR29	0000 0000$_H$	32	FFD0 07D4$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 29	RSCAN0RMDF029	0000 0000$_H$	32	FFD0 07D8$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 29	RSCAN0RMDF129	0000 0000$_H$	32	FFD0 07DC$_H$	2	8, 16, 32
RSCAN0	Receive Buffer ID Register 30	RSCAN0RMID30	0000 0000$_H$	32	FFD0 07E0$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 30	RSCAN0RMPTR30	0000 0000$_H$	32	FFD0 07E4$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 30	RSCAN0RMDF030	0000 0000$_H$	32	FFD0 07E8$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 30	RSCAN0RMDF130	0000 0000$_H$	32	FFD0 07EC$_H$	2	8, 16, 32
RSCAN0	Receive Buffer ID Register 31	RSCAN0RMID31	0000 0000$_H$	32	FFD0 07F0$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 31	RSCAN0RMPTR31	0000 0000$_H$	32	FFD0 07F4$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 31	RSCAN0RMDF031	0000 0000$_H$	32	FFD0 07F8$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 31	RSCAN0RMDF131	0000 0000$_H$	32	FFD0 07FC$_H$	2	8, 16, 32
RSCAN0	Receive Buffer ID Register 32	RSCAN0RMID32	0000 0000$_H$	32	FFD0 0800$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 32	RSCAN0RMPTR32	0000 0000$_H$	32	FFD0 0804$_H$	2	8, 16, 32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
RSCAN0	Receive Buffer Data Field 0 Register 32	RSCAN0RMDF032	0000 0000$_H$	32	FFD0 0808$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 32	RSCAN0RMDF132	0000 0000$_H$	32	FFD0 080C$_H$	2	8, 16, 32
RSCAN0	Receive Buffer ID Register 33	RSCAN0RMID33	0000 0000$_H$	32	FFD0 0810$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 33	RSCAN0RMPTR33	0000 0000$_H$	32	FFD0 0814$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 33	RSCAN0RMDF033	0000 0000$_H$	32	FFD0 0818$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 33	RSCAN0RMDF133	0000 0000$_H$	32	FFD0 081C$_H$	2	8, 16, 32
RSCAN0	Receive Buffer ID Register 34	RSCAN0RMID34	0000 0000$_H$	32	FFD0 0820$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 34	RSCAN0RMPTR34	0000 0000$_H$	32	FFD0 0824$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 34	RSCAN0RMDF034	0000 0000$_H$	32	FFD0 0828$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 34	RSCAN0RMDF134	0000 0000$_H$	32	FFD0 082C$_H$	2	8, 16, 32
RSCAN0	Receive Buffer ID Register 35	RSCAN0RMID35	0000 0000$_H$	32	FFD0 0830$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 35	RSCAN0RMPTR35	0000 0000$_H$	32	FFD0 0834$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 35	RSCAN0RMDF035	0000 0000$_H$	32	FFD0 0838$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 35	RSCAN0RMDF135	0000 0000$_H$	32	FFD0 083C$_H$	2	8, 16, 32
RSCAN0	Receive Buffer ID Register 36	RSCAN0RMID36	0000 0000$_H$	32	FFD0 0840$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 36	RSCAN0RMPTR36	0000 0000$_H$	32	FFD0 0844$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 36	RSCAN0RMDF036	0000 0000$_H$	32	FFD0 0848$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 36	RSCAN0RMDF136	0000 0000$_H$	32	FFD0 084C$_H$	2	8, 16, 32
RSCAN0	Receive Buffer ID Register 37	RSCAN0RMID37	0000 0000$_H$	32	FFD0 0850$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 37	RSCAN0RMPTR37	0000 0000$_H$	32	FFD0 0854$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 37	RSCAN0RMDF037	0000 0000$_H$	32	FFD0 0858$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 37	RSCAN0RMDF137	0000 0000$_H$	32	FFD0 085C$_H$	2	8, 16, 32
RSCAN0	Receive Buffer ID Register 38	RSCAN0RMID38	0000 0000$_H$	32	FFD0 0860$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 38	RSCAN0RMPTR38	0000 0000$_H$	32	FFD0 0864$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 38	RSCAN0RMDF038	0000 0000$_H$	32	FFD0 0868$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 38	RSCAN0RMDF138	0000 0000$_H$	32	FFD0 086C$_H$	2	8, 16, 32
RSCAN0	Receive Buffer ID Register 39	RSCAN0RMID39	0000 0000$_H$	32	FFD0 0870$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 39	RSCAN0RMPTR39	0000 0000$_H$	32	FFD0 0874$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 39	RSCAN0RMDF039	0000 0000$_H$	32	FFD0 0878$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 39	RSCAN0RMDF139	0000 0000$_H$	32	FFD0 087C$_H$	2	8, 16, 32
RSCAN0	Receive Buffer ID Register 40	RSCAN0RMID40	0000 0000$_H$	32	FFD0 0880$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 40	RSCAN0RMPTR40	0000 0000$_H$	32	FFD0 0884$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 40	RSCAN0RMDF040	0000 0000$_H$	32	FFD0 0888$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 40	RSCAN0RMDF140	0000 0000$_H$	32	FFD0 088C$_H$	2	8, 16, 32
RSCAN0	Receive Buffer ID Register 41	RSCAN0RMID41	0000 0000$_H$	32	FFD0 0890$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 41	RSCAN0RMPTR41	0000 0000$_H$	32	FFD0 0894$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 41	RSCAN0RMDF041	0000 0000$_H$	32	FFD0 0898$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 41	RSCAN0RMDF141	0000 0000$_H$	32	FFD0 089C$_H$	2	8, 16, 32
RSCAN0	Receive Buffer ID Register 42	RSCAN0RMID42	0000 0000$_H$	32	FFD0 08A0$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 42	RSCAN0RMPTR42	0000 0000$_H$	32	FFD0 08A4$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 42	RSCAN0RMDF042	0000 0000$_H$	32	FFD0 08A8$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 42	RSCAN0RMDF142	0000 0000$_H$	32	FFD0 08AC$_H$	2	8, 16, 32
RSCAN0	Receive Buffer ID Register 43	RSCAN0RMID43	0000 0000$_H$	32	FFD0 08B0$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 43	RSCAN0RMPTR43	0000 0000$_H$	32	FFD0 08B4$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 43	RSCAN0RMDF043	0000 0000$_H$	32	FFD0 08B8$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 43	RSCAN0RMDF143	0000 0000$_H$	32	FFD0 08BC$_H$	2	8, 16, 32
RSCAN0	Receive Buffer ID Register 44	RSCAN0RMID44	0000 0000$_H$	32	FFD0 08C0$_H$	2	8, 16, 32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
RSCAN0	Receive Buffer Pointer Register 44	RSCAN0RMPTR44	0000 0000$_H$	32	FFD0 08C4$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 44	RSCAN0RMDF044	0000 0000$_H$	32	FFD0 08C8$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 44	RSCAN0RMDF144	0000 0000$_H$	32	FFD0 08CC$_H$	2	8, 16, 32
RSCAN0	Receive Buffer ID Register 45	RSCAN0RMID45	0000 0000$_H$	32	FFD0 08D0$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 45	RSCAN0RMPTR45	0000 0000$_H$	32	FFD0 08D4$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 45	RSCAN0RMDF045	0000 0000$_H$	32	FFD0 08D8$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 45	RSCAN0RMDF145	0000 0000$_H$	32	FFD0 08DC$_H$	2	8, 16, 32
RSCAN0	Receive Buffer ID Register 46	RSCAN0RMID46	0000 0000$_H$	32	FFD0 08E0$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 46	RSCAN0RMPTR46	0000 0000$_H$	32	FFD0 08E4$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 46	RSCAN0RMDF046	0000 0000$_H$	32	FFD0 08E8$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 46	RSCAN0RMDF146	0000 0000$_H$	32	FFD0 08EC$_H$	2	8, 16, 32
RSCAN0	Receive Buffer ID Register 47	RSCAN0RMID47	0000 0000$_H$	32	FFD0 08F0$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 47	RSCAN0RMPTR47	0000 0000$_H$	32	FFD0 08F4$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 47	RSCAN0RMDF047	0000 0000$_H$	32	FFD0 08F8$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 47	RSCAN0RMDF147	0000 0000$_H$	32	FFD0 08FC$_H$	2	8, 16, 32
RSCAN0	Receive Buffer ID Register 48	RSCAN0RMID48	0000 0000$_H$	32	FFD0 0900$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 48	RSCAN0RMPTR48	0000 0000$_H$	32	FFD0 0904$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 48	RSCAN0RMDF048	0000 0000$_H$	32	FFD0 0908$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 48	RSCAN0RMDF148	0000 0000$_H$	32	FFD0 090C$_H$	2	8, 16, 32
RSCAN0	Receive Buffer ID Register 49	RSCAN0RMID49	0000 0000$_H$	32	FFD0 0910$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 49	RSCAN0RMPTR49	0000 0000$_H$	32	FFD0 0914$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 49	RSCAN0RMDF049	0000 0000$_H$	32	FFD0 0918$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 49	RSCAN0RMDF149	0000 0000$_H$	32	FFD0 091C$_H$	2	8, 16, 32
RSCAN0	Receive Buffer ID Register 50	RSCAN0RMID50	0000 0000$_H$	32	FFD0 0920$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 50	RSCAN0RMPTR50	0000 0000$_H$	32	FFD0 0924$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 50	RSCAN0RMDF050	0000 0000$_H$	32	FFD0 0928$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 50	RSCAN0RMDF150	0000 0000$_H$	32	FFD0 092C$_H$	2	8, 16, 32
RSCAN0	Receive Buffer ID Register 51	RSCAN0RMID51	0000 0000$_H$	32	FFD0 0930$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 51	RSCAN0RMPTR51	0000 0000$_H$	32	FFD0 0934$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 51	RSCAN0RMDF051	0000 0000$_H$	32	FFD0 0938$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 51	RSCAN0RMDF151	0000 0000$_H$	32	FFD0 093C$_H$	2	8, 16, 32
RSCAN0	Receive Buffer ID Register 52	RSCAN0RMID52	0000 0000$_H$	32	FFD0 0940$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 52	RSCAN0RMPTR52	0000 0000$_H$	32	FFD0 0944$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 52	RSCAN0RMDF052	0000 0000$_H$	32	FFD0 0948$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 52	RSCAN0RMDF152	0000 0000$_H$	32	FFD0 094C$_H$	2	8, 16, 32
RSCAN0	Receive Buffer ID Register 53	RSCAN0RMID53	0000 0000$_H$	32	FFD0 0950$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 53	RSCAN0RMPTR53	0000 0000$_H$	32	FFD0 0954$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 53	RSCAN0RMDF053	0000 0000$_H$	32	FFD0 0958$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 53	RSCAN0RMDF153	0000 0000$_H$	32	FFD0 095C$_H$	2	8, 16, 32
RSCAN0	Receive Buffer ID Register 54	RSCAN0RMID54	0000 0000$_H$	32	FFD0 0960$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 54	RSCAN0RMPTR54	0000 0000$_H$	32	FFD0 0964$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 54	RSCAN0RMDF054	0000 0000$_H$	32	FFD0 0968$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 54	RSCAN0RMDF154	0000 0000$_H$	32	FFD0 096C$_H$	2	8, 16, 32
RSCAN0	Receive Buffer ID Register 55	RSCAN0RMID55	0000 0000$_H$	32	FFD0 0970$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 55	RSCAN0RMPTR55	0000 0000$_H$	32	FFD0 0974$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 55	RSCAN0RMDF055	0000 0000$_H$	32	FFD0 0978$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 55	RSCAN0RMDF155	0000 0000$_H$	32	FFD0 097C$_H$	2	8, 16, 32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
RSCAN0	Receive Buffer ID Register 56	RSCAN0RMID56	0000 0000$_H$	32	FFD0 0980$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 56	RSCAN0RMPTR56	0000 0000$_H$	32	FFD0 0984$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 56	RSCAN0RMDF056	0000 0000$_H$	32	FFD0 0988$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 56	RSCAN0RMDF156	0000 0000$_H$	32	FFD0 098C$_H$	2	8, 16, 32
RSCAN0	Receive Buffer ID Register 57	RSCAN0RMID57	0000 0000$_H$	32	FFD0 0990$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 57	RSCAN0RMPTR57	0000 0000$_H$	32	FFD0 0994$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 57	RSCAN0RMDF057	0000 0000$_H$	32	FFD0 0998$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 57	RSCAN0RMDF157	0000 0000$_H$	32	FFD0 099C$_H$	2	8, 16, 32
RSCAN0	Receive Buffer ID Register 58	RSCAN0RMID58	0000 0000$_H$	32	FFD0 09A0$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 58	RSCAN0RMPTR58	0000 0000$_H$	32	FFD0 09A4$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 58	RSCAN0RMDF058	0000 0000$_H$	32	FFD0 09A8$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 58	RSCAN0RMDF158	0000 0000$_H$	32	FFD0 09AC$_H$	2	8, 16, 32
RSCAN0	Receive Buffer ID Register 59	RSCAN0RMID59	0000 0000$_H$	32	FFD0 09B0$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 59	RSCAN0RMPTR59	0000 0000$_H$	32	FFD0 09B4$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 59	RSCAN0RMDF059	0000 0000$_H$	32	FFD0 09B8$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 59	RSCAN0RMDF159	0000 0000$_H$	32	FFD0 09BC$_H$	2	8, 16, 32
RSCAN0	Receive Buffer ID Register 60	RSCAN0RMID60	0000 0000$_H$	32	FFD0 09C0$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 60	RSCAN0RMPTR60	0000 0000$_H$	32	FFD0 09C4$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 60	RSCAN0RMDF060	0000 0000$_H$	32	FFD0 09C8$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 60	RSCAN0RMDF160	0000 0000$_H$	32	FFD0 09CC$_H$	2	8, 16, 32
RSCAN0	Receive Buffer ID Register 61	RSCAN0RMID61	0000 0000$_H$	32	FFD0 09D0$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 61	RSCAN0RMPTR61	0000 0000$_H$	32	FFD0 09D4$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 61	RSCAN0RMDF061	0000 0000$_H$	32	FFD0 09D8$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 61	RSCAN0RMDF161	0000 0000$_H$	32	FFD0 09DC$_H$	2	8, 16, 32
RSCAN0	Receive Buffer ID Register 62	RSCAN0RMID62	0000 0000$_H$	32	FFD0 09E0$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 62	RSCAN0RMPTR62	0000 0000$_H$	32	FFD0 09E4$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 62	RSCAN0RMDF062	0000 0000$_H$	32	FFD0 09E8$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 62	RSCAN0RMDF162	0000 0000$_H$	32	FFD0 09EC$_H$	2	8, 16, 32
RSCAN0	Receive Buffer ID Register 63	RSCAN0RMID63	0000 0000$_H$	32	FFD0 09F0$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Pointer Register 63	RSCAN0RMPTR63	0000 0000$_H$	32	FFD0 09F4$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 0 Register 63	RSCAN0RMDF063	0000 0000$_H$	32	FFD0 09F8$_H$	2	8, 16, 32
RSCAN0	Receive Buffer Data Field 1 Register 63	RSCAN0RMDF163	0000 0000$_H$	32	FFD0 09FC$_H$	2	8, 16, 32
RSCAN0	Receive FIFO Buffer Access ID Register 0	RSCAN0RFID0	0000 0000$_H$	32	FFD0 0E00$_H$	2	8, 16, 32
RSCAN0	Receive FIFO Buffer Access Point Register 0	RSCAN0RFPTR0	0000 0000$_H$	32	FFD0 0E04$_H$	2	8, 16, 32
RSCAN0	Receive FIFO Buffer Access Data Field 0 Register 0	RSCAN0RFDF00	0000 0000$_H$	32	FFD0 0E08$_H$	2	8, 16, 32
RSCAN0	Receive FIFO Buffer Access Data Field 1 Register 0	RSCAN0RFDF10	0000 0000$_H$	32	FFD0 0E0C$_H$	2	8, 16, 32
RSCAN0	Receive FIFO Buffer Access ID Register 1	RSCAN0RFID1	0000 0000$_H$	32	FFD0 0E10$_H$	2	8, 16, 32
RSCAN0	Receive FIFO Buffer Access Point Register 1	RSCAN0RFPTR1	0000 0000$_H$	32	FFD0 0E14$_H$	2	8, 16, 32
RSCAN0	Receive FIFO Buffer Access Data Field 0 Register 1	RSCAN0RFDF01	0000 0000$_H$	32	FFD0 0E18$_H$	2	8, 16, 32
RSCAN0	Receive FIFO Buffer Access Data Field 1 Register 1	RSCAN0RFDF11	0000 0000$_H$	32	FFD0 0E1C$_H$	2	8, 16, 32
RSCAN0	Receive FIFO Buffer Access ID Register 2	RSCAN0RFID2	0000 0000$_H$	32	FFD0 0E20$_H$	2	8, 16, 32
RSCAN0	Receive FIFO Buffer Access Point Register 2	RSCAN0RFPTR2	0000 0000$_H$	32	FFD0 0E24$_H$	2	8, 16, 32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
RSCAN0	Receive FIFO Buffer Access Data Field 0 Register 2	RSCAN0RFDF02	0000 0000$_H$	32	FFD0 0E28$_H$	2	8, 16, 32
RSCAN0	Receive FIFO Buffer Access Data Field 1 Register 2	RSCAN0RFDF12	0000 0000$_H$	32	FFD0 0E2C$_H$	2	8, 16, 32
RSCAN0	Receive FIFO Buffer Access ID Register 3	RSCAN0RFID3	0000 0000$_H$	32	FFD0 0E30$_H$	2	8, 16, 32
RSCAN0	Receive FIFO Buffer Access Point Register 3	RSCAN0RFPTR3	0000 0000$_H$	32	FFD0 0E34$_H$	2	8, 16, 32
RSCAN0	Receive FIFO Buffer Access Data Field 0 Register 3	RSCAN0RFDF03	0000 0000$_H$	32	FFD0 0E38$_H$	2	8, 16, 32
RSCAN0	Receive FIFO Buffer Access Data Field 1 Register 3	RSCAN0RFDF13	0000 0000$_H$	32	FFD0 0E3C$_H$	2	8, 16, 32
RSCAN0	Receive FIFO Buffer Access ID Register 4	RSCAN0RFID4	0000 0000$_H$	32	FFD0 0E40$_H$	2	8, 16, 32
RSCAN0	Receive FIFO Buffer Access Point Register 4	RSCAN0RFPTR4	0000 0000$_H$	32	FFD0 0E44$_H$	2	8, 16, 32
RSCAN0	Receive FIFO Buffer Access Data Field 0 Register 4	RSCAN0RFDF04	0000 0000$_H$	32	FFD0 0E48$_H$	2	8, 16, 32
RSCAN0	Receive FIFO Buffer Access Data Field 1 Register 4	RSCAN0RFDF14	0000 0000$_H$	32	FFD0 0E4C$_H$	2	8, 16, 32
RSCAN0	Receive FIFO Buffer Access ID Register 5	RSCAN0RFID5	0000 0000$_H$	32	FFD0 0E50$_H$	2	8, 16, 32
RSCAN0	Receive FIFO Buffer Access Point Register 5	RSCAN0RFPTR5	0000 0000$_H$	32	FFD0 0E54$_H$	2	8, 16, 32
RSCAN0	Receive FIFO Buffer Access Data Field 0 Register 5	RSCAN0RFDF05	0000 0000$_H$	32	FFD0 0E58$_H$	2	8, 16, 32
RSCAN0	Receive FIFO Buffer Access Data Field 1 Register 5	RSCAN0RFDF15	0000 0000$_H$	32	FFD0 0E5C$_H$	2	8, 16, 32
RSCAN0	Receive FIFO Buffer Access ID Register 6	RSCAN0RFID6	0000 0000$_H$	32	FFD0 0E60$_H$	2	8, 16, 32
RSCAN0	Receive FIFO Buffer Access Point Register 6	RSCAN0RFPTR6	0000 0000$_H$	32	FFD0 0E64$_H$	2	8, 16, 32
RSCAN0	Receive FIFO Buffer Access Data Field 0 Register 6	RSCAN0RFDF06	0000 0000$_H$	32	FFD0 0E68$_H$	2	8, 16, 32
RSCAN0	Receive FIFO Buffer Access Data Field 1 Register 6	RSCAN0RFDF16	0000 0000$_H$	32	FFD0 0E6C$_H$	2	8, 16, 32
RSCAN0	Receive FIFO Buffer Access ID Register 7	RSCAN0RFID7	0000 0000$_H$	32	FFD0 0E70$_H$	2	8, 16, 32
RSCAN0	Receive FIFO Buffer Access Point Register 7	RSCAN0RFPTR7	0000 0000$_H$	32	FFD0 0E74$_H$	2	8, 16, 32
RSCAN0	Receive FIFO Buffer Access Data Field 0 Register 7	RSCAN0RFDF07	0000 0000$_H$	32	FFD0 0E78$_H$	2	8, 16, 32
RSCAN0	Receive FIFO Buffer Access Data Field 1 Register 7	RSCAN0RFDF17	0000 0000$_H$	32	FFD0 0E7C$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Access ID Register 0	RSCAN0CFID0	0000 0000$_H$	32	FFD0 0E80$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Access Point Register 0	RSCAN0CFPTR0	0000 0000$_H$	32	FFD0 0E84$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Access Data Field 0 Register 0	RSCAN0CFDF00	0000 0000$_H$	32	FFD0 0E88$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Access Data Field 1 Register 0	RSCAN0CFDF10	0000 0000$_H$	32	FFD0 0E8C$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Access ID Register 1	RSCAN0CFID1	0000 0000$_H$	32	FFD0 0E90$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Access Point Register 1	RSCAN0CFPTR1	0000 0000$_H$	32	FFD0 0E94$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Access Data Field 0 Register 1	RSCAN0CFDF01	0000 0000$_H$	32	FFD0 0E98$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Access Data Field 1 Register 1	RSCAN0CFDF11	0000 0000$_H$	32	FFD0 0E9C$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Access ID Register 2	RSCAN0CFID2	0000 0000$_H$	32	FFD0 0EA0$_H$	2	8, 16, 32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
RSCAN0	Transmit/receive FIFO Buffer Access Point Register 2	RSCAN0CFPTR2	0000 0000$_H$	32	FFD0 0EA4$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Access Data Field 0 Register 2	RSCAN0CFDF02	0000 0000$_H$	32	FFD0 0EA8$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Access Data Field 1 Register 2	RSCAN0CFDF12	0000 0000$_H$	32	FFD0 0EAC$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Access ID Register 3	RSCAN0CFID3	0000 0000$_H$	32	FFD0 0EB0$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Access Point Register 3	RSCAN0CFPTR3	0000 0000$_H$	32	FFD0 0EB4$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Access Data Field 0 Register 3	RSCAN0CFDF03	0000 0000$_H$	32	FFD0 0EB8$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Access Data Field 1 Register 3	RSCAN0CFDF13	0000 0000$_H$	32	FFD0 0EBC$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Access ID Register 4	RSCAN0CFID4	0000 0000$_H$	32	FFD0 0EC0$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Access Point Register 4	RSCAN0CFPTR4	0000 0000$_H$	32	FFD0 0EC4$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Access Data Field 0 Register 4	RSCAN0CFDF04	0000 0000$_H$	32	FFD0 0EC8$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Access Data Field 1 Register 4	RSCAN0CFDF14	0000 0000$_H$	32	FFD0 0ECC$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Access ID Register 5	RSCAN0CFID5	0000 0000$_H$	32	FFD0 0ED0$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Access Point Register 5	RSCAN0CFPTR5	0000 0000$_H$	32	FFD0 0ED4$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Access Data Field 0 Register 5	RSCAN0CFDF05	0000 0000$_H$	32	FFD0 0ED8$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Access Data Field 1 Register 5	RSCAN0CFDF15	0000 0000$_H$	32	FFD0 0EDC$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Access ID Register 6	RSCAN0CFID6	0000 0000$_H$	32	FFD0 0EE0$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Access Point Register 6	RSCAN0CFPTR6	0000 0000$_H$	32	FFD0 0EE4$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Access Data Field 0 Register 6	RSCAN0CFDF06	0000 0000$_H$	32	FFD0 0EE8$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Access Data Field 1 Register 6	RSCAN0CFDF16	0000 0000$_H$	32	FFD0 0EEC$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Access ID Register 7	RSCAN0CFID7	0000 0000$_H$	32	FFD0 0EF0$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Access Point Register 7	RSCAN0CFPTR7	0000 0000$_H$	32	FFD0 0EF4$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Access Data Field 0 Register 7	RSCAN0CFDF07	0000 0000$_H$	32	FFD0 0EF8$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Access Data Field 1 Register 7	RSCAN0CFDF17	0000 0000$_H$	32	FFD0 0EFC$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Access ID Register 8	RSCAN0CFID8	0000 0000$_H$	32	FFD0 0F00$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Access Point Register 8	RSCAN0CFPTR8	0000 0000$_H$	32	FFD0 0F04$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Access Data Field 0 Register 8	RSCAN0CFDF08	0000 0000$_H$	32	FFD0 0F08$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Access Data Field 1 Register 8	RSCAN0CFDF18	0000 0000$_H$	32	FFD0 0F0C$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Access ID Register 9	RSCAN0CFID9	0000 0000$_H$	32	FFD0 0F10$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Access Point Register 9	RSCAN0CFPTR9	0000 0000$_H$	32	FFD0 0F14$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Access Data Field 0 Register 9	RSCAN0CFDF09	0000 0000$_H$	32	FFD0 0F18$_H$	2	8, 16, 32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
RSCAN0	Transmit/receive FIFO Buffer Access Data Field 1 Register 9	RSCAN0CFDF19	0000 0000$_H$	32	FFD0 0F1C$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Access ID Register 10	RSCAN0CFID10	0000 0000$_H$	32	FFD0 0F20$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Access Point Register 10	RSCAN0CFPTR10	0000 0000$_H$	32	FFD0 0F24$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Access Data Field 0 Register 10	RSCAN0CFDF010	0000 0000$_H$	32	FFD0 0F28$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Access Data Field 1 Register 10	RSCAN0CFDF110	0000 0000$_H$	32	FFD0 0F2C$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Access ID Register 11	RSCAN0CFID11	0000 0000$_H$	32	FFD0 0F30$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Access Point Register 11	RSCAN0CFPTR11	0000 0000$_H$	32	FFD0 0F34$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Access Data Field 0 Register 11	RSCAN0CFDF011	0000 0000$_H$	32	FFD0 0F38$_H$	2	8, 16, 32
RSCAN0	Transmit/receive FIFO Buffer Access Data Field 1 Register 11	RSCAN0CFDF111	0000 0000$_H$	32	FFD0 0F3C$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 0	RSCAN0TMID0	0000 0000$_H$	32	FFD0 1000$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Pointer Register 0	RSCAN0TMPTR0	0000 0000$_H$	32	FFD0 1004$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 0	RSCAN0TMDF00	0000 0000$_H$	32	FFD0 1008$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 0	RSCAN0TMDF10	0000 0000$_H$	32	FFD0 100C$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 1	RSCAN0TMID1	0000 0000$_H$	32	FFD0 1010$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Pointer Register 1	RSCAN0TMPTR1	0000 0000$_H$	32	FFD0 1014$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 1	RSCAN0TMDF01	0000 0000$_H$	32	FFD0 1018$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 1	RSCAN0TMDF11	0000 0000$_H$	32	FFD0 101C$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 2	RSCAN0TMID2	0000 0000$_H$	32	FFD0 1020$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Pointer Register 2	RSCAN0TMPTR2	0000 0000$_H$	32	FFD0 1024$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 2	RSCAN0TMDF02	0000 0000$_H$	32	FFD0 1028$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 2	RSCAN0TMDF12	0000 0000$_H$	32	FFD0 102C$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 3	RSCAN0TMID3	0000 0000$_H$	32	FFD0 1030$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Pointer Register 3	RSCAN0TMPTR3	0000 0000$_H$	32	FFD0 1034$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 3	RSCAN0TMDF03	0000 0000$_H$	32	FFD0 1038$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 3	RSCAN0TMDF13	0000 0000$_H$	32	FFD0 103C$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 4	RSCAN0TMID4	0000 0000$_H$	32	FFD0 1040$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Pointer Register 4	RSCAN0TMPTR4	0000 0000$_H$	32	FFD0 1044$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 4	RSCAN0TMDF04	0000 0000$_H$	32	FFD0 1048$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 4	RSCAN0TMDF14	0000 0000$_H$	32	FFD0 104C$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 5	RSCAN0TMID5	0000 0000$_H$	32	FFD0 1050$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Pointer Register 5	RSCAN0TMPTR5	0000 0000$_H$	32	FFD0 1054$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 5	RSCAN0TMDF05	0000 0000$_H$	32	FFD0 1058$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 5	RSCAN0TMDF15	0000 0000$_H$	32	FFD0 105C$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 6	RSCAN0TMID6	0000 0000$_H$	32	FFD0 1060$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Pointer Register 6	RSCAN0TMPTR6	0000 0000$_H$	32	FFD0 1064$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 6	RSCAN0TMDF06	0000 0000$_H$	32	FFD0 1068$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 6	RSCAN0TMDF16	0000 0000$_H$	32	FFD0 106C$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 7	RSCAN0TMID7	0000 0000$_H$	32	FFD0 1070$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Pointer Register 7	RSCAN0TMPTR7	0000 0000$_H$	32	FFD0 1074$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 7	RSCAN0TMDF07	0000 0000$_H$	32	FFD0 1078$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 7	RSCAN0TMDF17	0000 0000$_H$	32	FFD0 107C$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 8	RSCAN0TMID8	0000 0000$_H$	32	FFD0 1080$_H$	2	8, 16, 32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
RSCAN0	Transfer Buffer Pointer Register 8	RSCAN0TMPTR8	0000 0000$_H$	32	FFD0 1084$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 8	RSCAN0TMDF08	0000 0000$_H$	32	FFD0 1088$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 8	RSCAN0TMDF18	0000 0000$_H$	32	FFD0 108C$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 9	RSCAN0TMID9	0000 0000$_H$	32	FFD0 1090$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Pointer Register 9	RSCAN0TMPTR9	0000 0000$_H$	32	FFD0 1094$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 9	RSCAN0TMDF09	0000 0000$_H$	32	FFD0 1098$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 9	RSCAN0TMDF19	0000 0000$_H$	32	FFD0 109C$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 10	RSCAN0TMID10	0000 0000$_H$	32	FFD0 10A0$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Pointer Register 10	RSCAN0TMPTR10	0000 0000$_H$	32	FFD0 10A4$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 10	RSCAN0TMDF010	0000 0000$_H$	32	FFD0 10A8$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 10	RSCAN0TMDF110	0000 0000$_H$	32	FFD0 10AC$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 11	RSCAN0TMID11	0000 0000$_H$	32	FFD0 10B0$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Pointer Register 11	RSCAN0TMPTR11	0000 0000$_H$	32	FFD0 10B4$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 11	RSCAN0TMDF011	0000 0000$_H$	32	FFD0 10B8$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 11	RSCAN0TMDF111	0000 0000$_H$	32	FFD0 10BC$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 12	RSCAN0TMID12	0000 0000$_H$	32	FFD0 10C0$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Pointer Register 12	RSCAN0TMPTR12	0000 0000$_H$	32	FFD0 10C4$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 12	RSCAN0TMDF012	0000 0000$_H$	32	FFD0 10C8$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 12	RSCAN0TMDF112	0000 0000$_H$	32	FFD0 10CC$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 13	RSCAN0TMID13	0000 0000$_H$	32	FFD0 10D0$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Pointer Register 13	RSCAN0TMPTR13	0000 0000$_H$	32	FFD0 10D4$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 13	RSCAN0TMDF013	0000 0000$_H$	32	FFD0 10D8$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 13	RSCAN0TMDF113	0000 0000$_H$	32	FFD0 10DC$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 14	RSCAN0TMID14	0000 0000$_H$	32	FFD0 10E0$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Pointer Register 14	RSCAN0TMPTR14	0000 0000$_H$	32	FFD0 10E4$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 14	RSCAN0TMDF014	0000 0000$_H$	32	FFD0 10E8$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 14	RSCAN0TMDF114	0000 0000$_H$	32	FFD0 10EC$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 15	RSCAN0TMID15	0000 0000$_H$	32	FFD0 10F0$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Pointer Register 15	RSCAN0TMPTR15	0000 0000$_H$	32	FFD0 10F4$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 15	RSCAN0TMDF015	0000 0000$_H$	32	FFD0 10F8$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 15	RSCAN0TMDF115	0000 0000$_H$	32	FFD0 10FC$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 16	RSCAN0TMID16	0000 0000$_H$	32	FFD0 1100$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Pointer Register 16	RSCAN0TMPTR16	0000 0000$_H$	32	FFD0 1104$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 16	RSCAN0TMDF016	0000 0000$_H$	32	FFD0 1108$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 16	RSCAN0TMDF116	0000 0000$_H$	32	FFD0 110C$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 17	RSCAN0TMID17	0000 0000$_H$	32	FFD0 1110$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Pointer Register 17	RSCAN0TMPTR17	0000 0000$_H$	32	FFD0 1114$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 17	RSCAN0TMDF017	0000 0000$_H$	32	FFD0 1118$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 17	RSCAN0TMDF117	0000 0000$_H$	32	FFD0 111C$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 18	RSCAN0TMID18	0000 0000$_H$	32	FFD0 1120$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Pointer Register 18	RSCAN0TMPTR18	0000 0000$_H$	32	FFD0 1124$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 18	RSCAN0TMDF018	0000 0000$_H$	32	FFD0 1128$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 18	RSCAN0TMDF118	0000 0000$_H$	32	FFD0 112C$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 19	RSCAN0TMID19	0000 0000$_H$	32	FFD0 1130$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Pointer Register 19	RSCAN0TMPTR19	0000 0000$_H$	32	FFD0 1134$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 19	RSCAN0TMDF019	0000 0000$_H$	32	FFD0 1138$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 19	RSCAN0TMDF119	0000 0000$_H$	32	FFD0 113C$_H$	2	8, 16, 32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
RSCAN0	Transfer Buffer ID Register 20	RSCAN0TMID20	0000 0000$_H$	32	FFD0 1140$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Pointer Register 20	RSCAN0TMPTR20	0000 0000$_H$	32	FFD0 1144$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 20	RSCAN0TMDF020	0000 0000$_H$	32	FFD0 1148$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 20	RSCAN0TMDF120	0000 0000$_H$	32	FFD0 114C$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 21	RSCAN0TMID21	0000 0000$_H$	32	FFD0 1150$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Pointer Register 21	RSCAN0TMPTR21	0000 0000$_H$	32	FFD0 1154$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 21	RSCAN0TMDF021	0000 0000$_H$	32	FFD0 1158$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 21	RSCAN0TMDF121	0000 0000$_H$	32	FFD0 115C$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 22	RSCAN0TMID22	0000 0000$_H$	32	FFD0 1160$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Pointer Register 22	RSCAN0TMPTR22	0000 0000$_H$	32	FFD0 1164$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 22	RSCAN0TMDF022	0000 0000$_H$	32	FFD0 1168$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 22	RSCAN0TMDF122	0000 0000$_H$	32	FFD0 116C$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 23	RSCAN0TMID23	0000 0000$_H$	32	FFD0 1170$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Pointer Register 23	RSCAN0TMPTR23	0000 0000$_H$	32	FFD0 1174$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 23	RSCAN0TMDF023	0000 0000$_H$	32	FFD0 1178$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 23	RSCAN0TMDF123	0000 0000$_H$	32	FFD0 117C$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 24	RSCAN0TMID24	0000 0000$_H$	32	FFD0 1180$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Pointer Register 24	RSCAN0TMPTR24	0000 0000$_H$	32	FFD0 1184$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 24	RSCAN0TMDF024	0000 0000$_H$	32	FFD0 1188$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 24	RSCAN0TMDF124	0000 0000$_H$	32	FFD0 118C$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 25	RSCAN0TMID25	0000 0000$_H$	32	FFD0 1190$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Pointer Register 25	RSCAN0TMPTR25	0000 0000$_H$	32	FFD0 1194$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 25	RSCAN0TMDF025	0000 0000$_H$	32	FFD0 1198$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 25	RSCAN0TMDF125	0000 0000$_H$	32	FFD0 119C$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 26	RSCAN0TMID26	0000 0000$_H$	32	FFD0 11A0$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Pointer Register 26	RSCAN0TMPTR26	0000 0000$_H$	32	FFD0 11A4$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 26	RSCAN0TMDF026	0000 0000$_H$	32	FFD0 11A8$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 26	RSCAN0TMDF126	0000 0000$_H$	32	FFD0 11AC$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 27	RSCAN0TMID27	0000 0000$_H$	32	FFD0 11B0$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Pointer Register 27	RSCAN0TMPTR27	0000 0000$_H$	32	FFD0 11B4$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 27	RSCAN0TMDF027	0000 0000$_H$	32	FFD0 11B8$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 27	RSCAN0TMDF127	0000 0000$_H$	32	FFD0 11BC$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 28	RSCAN0TMID28	0000 0000$_H$	32	FFD0 11C0$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Pointer Register 28	RSCAN0TMPTR28	0000 0000$_H$	32	FFD0 11C4$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 28	RSCAN0TMDF028	0000 0000$_H$	32	FFD0 11C8$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 28	RSCAN0TMDF128	0000 0000$_H$	32	FFD0 11CC$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 29	RSCAN0TMID29	0000 0000$_H$	32	FFD0 11D0$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Pointer Register 29	RSCAN0TMPTR29	0000 0000$_H$	32	FFD0 11D4$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 29	RSCAN0TMDF029	0000 0000$_H$	32	FFD0 11D8$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 29	RSCAN0TMDF129	0000 0000$_H$	32	FFD0 11DC$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 30	RSCAN0TMID30	0000 0000$_H$	32	FFD0 11E0$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Pointer Register 30	RSCAN0TMPTR30	0000 0000$_H$	32	FFD0 11E4$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 30	RSCAN0TMDF030	0000 0000$_H$	32	FFD0 11E8$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 30	RSCAN0TMDF130	0000 0000$_H$	32	FFD0 11EC$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 31	RSCAN0TMID31	0000 0000$_H$	32	FFD0 11F0$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Pointer Register 31	RSCAN0TMPTR31	0000 0000$_H$	32	FFD0 11F4$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 31	RSCAN0TMDF031	0000 0000$_H$	32	FFD0 11F8$_H$	2	8, 16, 32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
RSCAN0	Transfer Buffer Data Field 1 Register 31	RSCAN0TMDF131	0000 0000$_H$	32	FFD0 11FC$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 32	RSCAN0TMID32	0000 0000$_H$	32	FFD0 1200$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Pointer Register 32	RSCAN0TMPTR32	0000 0000$_H$	32	FFD0 1204$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 32	RSCAN0TMDF032	0000 0000$_H$	32	FFD0 1208$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 32	RSCAN0TMDF132	0000 0000$_H$	32	FFD0 120C$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 33	RSCAN0TMID33	0000 0000$_H$	32	FFD0 1210$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Pointer Register 33	RSCAN0TMPTR33	0000 0000$_H$	32	FFD0 1214$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 33	RSCAN0TMDF033	0000 0000$_H$	32	FFD0 1218$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 33	RSCAN0TMDF133	0000 0000$_H$	32	FFD0 121C$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 34	RSCAN0TMID34	0000 0000$_H$	32	FFD0 1220$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Pointer Register 34	RSCAN0TMPTR34	0000 0000$_H$	32	FFD0 1224$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 34	RSCAN0TMDF034	0000 0000$_H$	32	FFD0 1228$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 34	RSCAN0TMDF134	0000 0000$_H$	32	FFD0 122C$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 35	RSCAN0TMID35	0000 0000$_H$	32	FFD0 1230$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Pointer Register 35	RSCAN0TMPTR35	0000 0000$_H$	32	FFD0 1234$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 35	RSCAN0TMDF035	0000 0000$_H$	32	FFD0 1238$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 35	RSCAN0TMDF135	0000 0000$_H$	32	FFD0 123C$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 36	RSCAN0TMID36	0000 0000$_H$	32	FFD0 1240$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Pointer Register 36	RSCAN0TMPTR36	0000 0000$_H$	32	FFD0 1244$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 36	RSCAN0TMDF036	0000 0000$_H$	32	FFD0 1248$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 36	RSCAN0TMDF136	0000 0000$_H$	32	FFD0 124C$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 37	RSCAN0TMID37	0000 0000$_H$	32	FFD0 1250$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Pointer Register 37	RSCAN0TMPTR37	0000 0000$_H$	32	FFD0 1254$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 37	RSCAN0TMDF037	0000 0000$_H$	32	FFD0 1258$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 37	RSCAN0TMDF137	0000 0000$_H$	32	FFD0 125C$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 38	RSCAN0TMID38	0000 0000$_H$	32	FFD0 1260$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Pointer Register 38	RSCAN0TMPTR38	0000 0000$_H$	32	FFD0 1264$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 38	RSCAN0TMDF038	0000 0000$_H$	32	FFD0 1268$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 38	RSCAN0TMDF138	0000 0000$_H$	32	FFD0 126C$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 39	RSCAN0TMID39	0000 0000$_H$	32	FFD0 1270$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Pointer Register 39	RSCAN0TMPTR39	0000 0000$_H$	32	FFD0 1274$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 39	RSCAN0TMDF039	0000 0000$_H$	32	FFD0 1278$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 39	RSCAN0TMDF139	0000 0000$_H$	32	FFD0 127C$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 40	RSCAN0TMID40	0000 0000$_H$	32	FFD0 1280$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Pointer Register 40	RSCAN0TMPTR40	0000 0000$_H$	32	FFD0 1284$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 40	RSCAN0TMDF040	0000 0000$_H$	32	FFD0 1288$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 40	RSCAN0TMDF140	0000 0000$_H$	32	FFD0 128C$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 41	RSCAN0TMID41	0000 0000$_H$	32	FFD0 1290$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Pointer Register 41	RSCAN0TMPTR41	0000 0000$_H$	32	FFD0 1294$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 41	RSCAN0TMDF041	0000 0000$_H$	32	FFD0 1298$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 41	RSCAN0TMDF141	0000 0000$_H$	32	FFD0 129C$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 42	RSCAN0TMID42	0000 0000$_H$	32	FFD0 12A0$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Pointer Register 42	RSCAN0TMPTR42	0000 0000$_H$	32	FFD0 12A4$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 42	RSCAN0TMDF042	0000 0000$_H$	32	FFD0 12A8$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 42	RSCAN0TMDF142	0000 0000$_H$	32	FFD0 12AC$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 43	RSCAN0TMID43	0000 0000$_H$	32	FFD0 12B0$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Pointer Register 43	RSCAN0TMPTR43	0000 0000$_H$	32	FFD0 12B4$_H$	2	8, 16, 32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
RSCAN0	Transfer Buffer Data Field 0 Register 43	RSCAN0TMDF043	0000 0000$_H$	32	FFD0 12B8$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 43	RSCAN0TMDF143	0000 0000$_H$	32	FFD0 12BC$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 44	RSCAN0TMID44	0000 0000$_H$	32	FFD0 12C0$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Pointer Register 44	RSCAN0TMPTR44	0000 0000$_H$	32	FFD0 12C4$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 44	RSCAN0TMDF044	0000 0000$_H$	32	FFD0 12C8$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 44	RSCAN0TMDF144	0000 0000$_H$	32	FFD0 12CC$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 45	RSCAN0TMID45	0000 0000$_H$	32	FFD0 12D0$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Pointer Register 45	RSCAN0TMPTR45	0000 0000$_H$	32	FFD0 12D4$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 45	RSCAN0TMDF045	0000 0000$_H$	32	FFD0 12D8$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 45	RSCAN0TMDF145	0000 0000$_H$	32	FFD0 12DC$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 46	RSCAN0TMID46	0000 0000$_H$	32	FFD0 12E0$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Pointer Register 46	RSCAN0TMPTR46	0000 0000$_H$	32	FFD0 12E4$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 46	RSCAN0TMDF046	0000 0000$_H$	32	FFD0 12E8$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 46	RSCAN0TMDF146	0000 0000$_H$	32	FFD0 12EC$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 47	RSCAN0TMID47	0000 0000$_H$	32	FFD0 12F0$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Pointer Register 47	RSCAN0TMPTR47	0000 0000$_H$	32	FFD0 12F4$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 47	RSCAN0TMDF047	0000 0000$_H$	32	FFD0 12F8$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 47	RSCAN0TMDF147	0000 0000$_H$	32	FFD0 12FC$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 48	RSCAN0TMID48	0000 0000$_H$	32	FFD0 1300$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Pointer Register 48	RSCAN0TMPTR48	0000 0000$_H$	32	FFD0 1304$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 48	RSCAN0TMDF048	0000 0000$_H$	32	FFD0 1308$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 48	RSCAN0TMDF148	0000 0000$_H$	32	FFD0 130C$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 49	RSCAN0TMID49	0000 0000$_H$	32	FFD0 1310$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Pointer Register 49	RSCAN0TMPTR49	0000 0000$_H$	32	FFD0 1314$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 49	RSCAN0TMDF049	0000 0000$_H$	32	FFD0 1318$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 49	RSCAN0TMDF149	0000 0000$_H$	32	FFD0 131C$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 50	RSCAN0TMID50	0000 0000$_H$	32	FFD0 1320$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Pointer Register 50	RSCAN0TMPTR50	0000 0000$_H$	32	FFD0 1324$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 50	RSCAN0TMDF050	0000 0000$_H$	32	FFD0 1328$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 50	RSCAN0TMDF150	0000 0000$_H$	32	FFD0 132C$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 51	RSCAN0TMID51	0000 0000$_H$	32	FFD0 1330$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Pointer Register 51	RSCAN0TMPTR51	0000 0000$_H$	32	FFD0 1334$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 51	RSCAN0TMDF051	0000 0000$_H$	32	FFD0 1338$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 51	RSCAN0TMDF151	0000 0000$_H$	32	FFD0 133C$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 52	RSCAN0TMID52	0000 0000$_H$	32	FFD0 1340$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Pointer Register 52	RSCAN0TMPTR52	0000 0000$_H$	32	FFD0 1344$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 52	RSCAN0TMDF052	0000 0000$_H$	32	FFD0 1348$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 52	RSCAN0TMDF152	0000 0000$_H$	32	FFD0 134C$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 53	RSCAN0TMID53	0000 0000$_H$	32	FFD0 1350$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Pointer Register 53	RSCAN0TMPTR53	0000 0000$_H$	32	FFD0 1354$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 53	RSCAN0TMDF053	0000 0000$_H$	32	FFD0 1358$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 53	RSCAN0TMDF153	0000 0000$_H$	32	FFD0 135C$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 54	RSCAN0TMID54	0000 0000$_H$	32	FFD0 1360$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Pointer Register 54	RSCAN0TMPTR54	0000 0000$_H$	32	FFD0 1364$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 54	RSCAN0TMDF054	0000 0000$_H$	32	FFD0 1368$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 54	RSCAN0TMDF154	0000 0000$_H$	32	FFD0 136C$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 55	RSCAN0TMID55	0000 0000$_H$	32	FFD0 1370$_H$	2	8, 16, 32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
RSCAN0	Transfer Buffer Pointer Register 55	RSCAN0TMPTR55	0000 0000$_H$	32	FFD0 1374$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 55	RSCAN0TMDF055	0000 0000$_H$	32	FFD0 1378$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 55	RSCAN0TMDF155	0000 0000$_H$	32	FFD0 137C$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 56	RSCAN0TMID56	0000 0000$_H$	32	FFD0 1380$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Pointer Register 56	RSCAN0TMPTR56	0000 0000$_H$	32	FFD0 1384$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 56	RSCAN0TMDF056	0000 0000$_H$	32	FFD0 1388$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 56	RSCAN0TMDF156	0000 0000$_H$	32	FFD0 138C$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 57	RSCAN0TMID57	0000 0000$_H$	32	FFD0 1390$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Pointer Register 57	RSCAN0TMPTR57	0000 0000$_H$	32	FFD0 1394$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 57	RSCAN0TMDF057	0000 0000$_H$	32	FFD0 1398$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 57	RSCAN0TMDF157	0000 0000$_H$	32	FFD0 139C$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 58	RSCAN0TMID58	0000 0000$_H$	32	FFD0 13A0$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Pointer Register 58	RSCAN0TMPTR58	0000 0000$_H$	32	FFD0 13A4$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 58	RSCAN0TMDF058	0000 0000$_H$	32	FFD0 13A8$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 58	RSCAN0TMDF158	0000 0000$_H$	32	FFD0 13AC$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 59	RSCAN0TMID59	0000 0000$_H$	32	FFD0 13B0$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Pointer Register 59	RSCAN0TMPTR59	0000 0000$_H$	32	FFD0 13B4$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 59	RSCAN0TMDF059	0000 0000$_H$	32	FFD0 13B8$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 59	RSCAN0TMDF159	0000 0000$_H$	32	FFD0 13BC$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 60	RSCAN0TMID60	0000 0000$_H$	32	FFD0 13C0$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Pointer Register 60	RSCAN0TMPTR60	0000 0000$_H$	32	FFD0 13C4$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 60	RSCAN0TMDF060	0000 0000$_H$	32	FFD0 13C8$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 60	RSCAN0TMDF160	0000 0000$_H$	32	FFD0 13CC$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 61	RSCAN0TMID61	0000 0000$_H$	32	FFD0 13D0$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Pointer Register 61	RSCAN0TMPTR61	0000 0000$_H$	32	FFD0 13D4$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 61	RSCAN0TMDF061	0000 0000$_H$	32	FFD0 13D8$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 61	RSCAN0TMDF161	0000 0000$_H$	32	FFD0 13DC$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 62	RSCAN0TMID62	0000 0000$_H$	32	FFD0 13E0$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Pointer Register 62	RSCAN0TMPTR62	0000 0000$_H$	32	FFD0 13E4$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 62	RSCAN0TMDF062	0000 0000$_H$	32	FFD0 13E8$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 62	RSCAN0TMDF162	0000 0000$_H$	32	FFD0 13EC$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer ID Register 63	RSCAN0TMID63	0000 0000$_H$	32	FFD0 13F0$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Pointer Register 63	RSCAN0TMPTR63	0000 0000$_H$	32	FFD0 13F4$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 0 Register 63	RSCAN0TMDF063	0000 0000$_H$	32	FFD0 13F8$_H$	2	8, 16, 32
RSCAN0	Transfer Buffer Data Field 1 Register 63	RSCAN0TMDF163	0000 0000$_H$	32	FFD0 13FC$_H$	2	8, 16, 32
RSCAN0	Transfer history Access Register 0	RSCAN0THLACC0	0000 0000$_H$	32	FFD0 1800$_H$	2	8, 16, 32
RSCAN0	Transfer history Access Register 1	RSCAN0THLACC1	0000 0000$_H$	32	FFD0 1804$_H$	2	8, 16, 32
RSCAN0	Transfer history Access Register 2	RSCAN0THLACC2	0000 0000$_H$	32	FFD0 1808$_H$	2	8, 16, 32
RSCAN0	Transfer history Access Register 3	RSCAN0THLACC3	0000 0000$_H$	32	FFD0 180C$_H$	2	8, 16, 32
RSCAN0	RAM Test Page Access Register 0	RSCAN0RPGACC0	0000 0000$_H$	32	FFD0 1900$_H$	2	8, 16, 32
RSCAN0	RAM Test Page Access Register 1	RSCAN0RPGACC1	0000 0000$_H$	32	FFD0 1904$_H$	2	8, 16, 32
RSCAN0	RAM Test Page Access Register 2	RSCAN0RPGACC2	0000 0000$_H$	32	FFD0 1908$_H$	2	8, 16, 32
RSCAN0	RAM Test Page Access Register 3	RSCAN0RPGACC3	0000 0000$_H$	32	FFD0 190C$_H$	2	8, 16, 32
RSCAN0	RAM Test Page Access Register 4	RSCAN0RPGACC4	0000 0000$_H$	32	FFD0 1910$_H$	2	8, 16, 32
RSCAN0	RAM Test Page Access Register 5	RSCAN0RPGACC5	0000 0000$_H$	32	FFD0 1914$_H$	2	8, 16, 32
RSCAN0	RAM Test Page Access Register 6	RSCAN0RPGACC6	0000 0000$_H$	32	FFD0 1918$_H$	2	8, 16, 32
RSCAN0	RAM Test Page Access Register 7	RSCAN0RPGACC7	0000 0000$_H$	32	FFD0 191C$_H$	2	8, 16, 32

(61/201)

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
RSCAN0	RAM Test Page Access Register 8	RSCAN0RPGACC8	0000 0000$_H$	32	FFD0 1920$_H$	2	8, 16, 32
RSCAN0	RAM Test Page Access Register 9	RSCAN0RPGACC9	0000 0000$_H$	32	FFD0 1924$_H$	2	8, 16, 32
RSCAN0	RAM Test Page Access Register 10	RSCAN0RPGACC10	0000 0000$_H$	32	FFD0 1928$_H$	2	8, 16, 32
RSCAN0	RAM Test Page Access Register 11	RSCAN0RPGACC11	0000 0000$_H$	32	FFD0 192C$_H$	2	8, 16, 32
RSCAN0	RAM Test Page Access Register 12	RSCAN0RPGACC12	0000 0000$_H$	32	FFD0 1930$_H$	2	8, 16, 32
RSCAN0	RAM Test Page Access Register 13	RSCAN0RPGACC13	0000 0000$_H$	32	FFD0 1934$_H$	2	8, 16, 32
RSCAN0	RAM Test Page Access Register 14	RSCAN0RPGACC14	0000 0000$_H$	32	FFD0 1938$_H$	2	8, 16, 32
RSCAN0	RAM Test Page Access Register 15	RSCAN0RPGACC15	0000 0000$_H$	32	FFD0 193C$_H$	2	8, 16, 32
RSCAN0	RAM Test Page Access Register 16	RSCAN0RPGACC16	0000 0000$_H$	32	FFD0 1940$_H$	2	8, 16, 32
RSCAN0	RAM Test Page Access Register 17	RSCAN0RPGACC17	0000 0000$_H$	32	FFD0 1944$_H$	2	8, 16, 32
RSCAN0	RAM Test Page Access Register 18	RSCAN0RPGACC18	0000 0000$_H$	32	FFD0 1948$_H$	2	8, 16, 32
RSCAN0	RAM Test Page Access Register 19	RSCAN0RPGACC19	0000 0000$_H$	32	FFD0 194C$_H$	2	8, 16, 32
RSCAN0	RAM Test Page Access Register 20	RSCAN0RPGACC20	0000 0000$_H$	32	FFD0 1950$_H$	2	8, 16, 32
RSCAN0	RAM Test Page Access Register 21	RSCAN0RPGACC21	0000 0000$_H$	32	FFD0 1954$_H$	2	8, 16, 32
RSCAN0	RAM Test Page Access Register 22	RSCAN0RPGACC22	0000 0000$_H$	32	FFD0 1958$_H$	2	8, 16, 32
RSCAN0	RAM Test Page Access Register 23	RSCAN0RPGACC23	0000 0000$_H$	32	FFD0 195C$_H$	2	8, 16, 32
RSCAN0	RAM Test Page Access Register 24	RSCAN0RPGACC24	0000 0000$_H$	32	FFD0 1960$_H$	2	8, 16, 32
RSCAN0	RAM Test Page Access Register 25	RSCAN0RPGACC25	0000 0000$_H$	32	FFD0 1964$_H$	2	8, 16, 32
RSCAN0	RAM Test Page Access Register 26	RSCAN0RPGACC26	0000 0000$_H$	32	FFD0 1968$_H$	2	8, 16, 32
RSCAN0	RAM Test Page Access Register 27	RSCAN0RPGACC27	0000 0000$_H$	32	FFD0 196C$_H$	2	8, 16, 32
RSCAN0	RAM Test Page Access Register 28	RSCAN0RPGACC28	0000 0000$_H$	32	FFD0 1970$_H$	2	8, 16, 32
RSCAN0	RAM Test Page Access Register 29	RSCAN0RPGACC29	0000 0000$_H$	32	FFD0 1974$_H$	2	8, 16, 32
RSCAN0	RAM Test Page Access Register 30	RSCAN0RPGACC30	0000 0000$_H$	32	FFD0 1978$_H$	2	8, 16, 32
RSCAN0	RAM Test Page Access Register 31	RSCAN0RPGACC31	0000 0000$_H$	32	FFD0 197C$_H$	2	8, 16, 32
RSCAN0	RAM Test Page Access Register 32	RSCAN0RPGACC32	0000 0000$_H$	32	FFD0 1980$_H$	2	8, 16, 32
RSCAN0	RAM Test Page Access Register 33	RSCAN0RPGACC33	0000 0000$_H$	32	FFD0 1984$_H$	2	8, 16, 32
RSCAN0	RAM Test Page Access Register 34	RSCAN0RPGACC34	0000 0000$_H$	32	FFD0 1988$_H$	2	8, 16, 32
RSCAN0	RAM Test Page Access Register 35	RSCAN0RPGACC35	0000 0000$_H$	32	FFD0 198C$_H$	2	8, 16, 32
RSCAN0	RAM Test Page Access Register 36	RSCAN0RPGACC36	0000 0000$_H$	32	FFD0 1990$_H$	2	8, 16, 32
RSCAN0	RAM Test Page Access Register 37	RSCAN0RPGACC37	0000 0000$_H$	32	FFD0 1994$_H$	2	8, 16, 32
RSCAN0	RAM Test Page Access Register 38	RSCAN0RPGACC38	0000 0000$_H$	32	FFD0 1998$_H$	2	8, 16, 32
RSCAN0	RAM Test Page Access Register 39	RSCAN0RPGACC39	0000 0000$_H$	32	FFD0 199C$_H$	2	8, 16, 32
RSCAN0	RAM Test Page Access Register 40	RSCAN0RPGACC40	0000 0000$_H$	32	FFD0 19A0$_H$	2	8, 16, 32
RSCAN0	RAM Test Page Access Register 41	RSCAN0RPGACC41	0000 0000$_H$	32	FFD0 19A4$_H$	2	8, 16, 32
RSCAN0	RAM Test Page Access Register 42	RSCAN0RPGACC42	0000 0000$_H$	32	FFD0 19A8$_H$	2	8, 16, 32
RSCAN0	RAM Test Page Access Register 43	RSCAN0RPGACC43	0000 0000$_H$	32	FFD0 19AC$_H$	2	8, 16, 32
RSCAN0	RAM Test Page Access Register 44	RSCAN0RPGACC44	0000 0000$_H$	32	FFD0 19B0$_H$	2	8, 16, 32
RSCAN0	RAM Test Page Access Register 45	RSCAN0RPGACC45	0000 0000$_H$	32	FFD0 19B4$_H$	2	8, 16, 32
RSCAN0	RAM Test Page Access Register 46	RSCAN0RPGACC46	0000 0000$_H$	32	FFD0 19B8$_H$	2	8, 16, 32
RSCAN0	RAM Test Page Access Register 47	RSCAN0RPGACC47	0000 0000$_H$	32	FFD0 19BC$_H$	2	8, 16, 32
RSCAN0	RAM Test Page Access Register 48	RSCAN0RPGACC48	0000 0000$_H$	32	FFD0 19C0$_H$	2	8, 16, 32
RSCAN0	RAM Test Page Access Register 49	RSCAN0RPGACC49	0000 0000$_H$	32	FFD0 19C4$_H$	2	8, 16, 32
RSCAN0	RAM Test Page Access Register 50	RSCAN0RPGACC50	0000 0000$_H$	32	FFD0 19C8$_H$	2	8, 16, 32
RSCAN0	RAM Test Page Access Register 51	RSCAN0RPGACC51	0000 0000$_H$	32	FFD0 19CC$_H$	2	8, 16, 32
RSCAN0	RAM Test Page Access Register 52	RSCAN0RPGACC52	0000 0000$_H$	32	FFD0 19D0$_H$	2	8, 16, 32
RSCAN0	RAM Test Page Access Register 53	RSCAN0RPGACC53	0000 0000$_H$	32	FFD0 19D4$_H$	2	8, 16, 32
RSCAN0	RAM Test Page Access Register 54	RSCAN0RPGACC54	0000 0000$_H$	32	FFD0 19D8$_H$	2	8, 16, 32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
RSCAN0	RAM Test Page Access Register 55	RSCAN0RPGACC55	0000 0000$_H$	32	FFD0 19DC$_H$	2	8, 16, 32
RSCAN0	RAM Test Page Access Register 56	RSCAN0RPGACC56	0000 0000$_H$	32	FFD0 19E0$_H$	2	8, 16, 32
RSCAN0	RAM Test Page Access Register 57	RSCAN0RPGACC57	0000 0000$_H$	32	FFD0 19E4$_H$	2	8, 16, 32
RSCAN0	RAM Test Page Access Register 58	RSCAN0RPGACC58	0000 0000$_H$	32	FFD0 19E8$_H$	2	8, 16, 32
RSCAN0	RAM Test Page Access Register 59	RSCAN0RPGACC59	0000 0000$_H$	32	FFD0 19EC$_H$	2	8, 16, 32
RSCAN0	RAM Test Page Access Register 60	RSCAN0RPGACC60	0000 0000$_H$	32	FFD0 19F0$_H$	2	8, 16, 32
RSCAN0	RAM Test Page Access Register 61	RSCAN0RPGACC61	0000 0000$_H$	32	FFD0 19F4$_H$	2	8, 16, 32
RSCAN0	RAM Test Page Access Register 62	RSCAN0RPGACC62	0000 0000$_H$	32	FFD0 19F8$_H$	2	8, 16, 32
RSCAN0	RAM Test Page Access Register 63	RSCAN0RPGACC63	0000 0000$_H$	32	FFD0 19FC$_H$	2	8, 16, 32
CSIH0	CSIH0 Control Register 0	CSIH0CTL0	00$_H$	8	FFD8 0000$_H$	2	8
CSIH0	CSIH0 Status Register 0	CSIH0STR0	0000 0010$_H$	32	FFD8 0004$_H$	2	32
CSIH0	CSIH0 Status Clear Register 0	CSIH0STCR0	0000$_H$	16	FFD8 0008$_H$	2	16
CSIH0	CSIH0 Control Register 1	CSIH0CTL1	0000 0000$_H$	32	FFD8 0010$_H$	2	32
CSIH0	CSIH0 Control Register 2	CSIH0CTL2	E000$_H$	16	FFD8 0014$_H$	2	16
CSIH0	CSIH0 Transmission Data Register 0 for Word Access	CSIH0TX0W	XXXX XXXX$_H$	32	FFD8 1008$_H$	2	32
CSIH0	CSIH0 Transmission Data for half Word Access Register 0	CSIH0TX0H	XXXX$_H$	16	FFD8 100C$_H$	2	16
CSIH0	CSIH0 Receive Data Register 0 for Word Access	CSIH0RX0W	XXXX XXXX$_H$	32	FFD8 1010$_H$	2	32
CSIH0	CSIH0 Receive Data Register for Half Word Access 0	CSIH0RX0H	XXXX$_H$	16	FFD8 1014$_H$	2	16
CSIH0	CSIH0 Configuration Register 0	CSIH0CFG0	0000 0000$_H$	32	FFD8 1044$_H$	2	32
CSIH0	CSIH0 Configuration Register 1	CSIH0CFG1	0000 0000$_H$	32	FFD8 1048$_H$	2	32
CSIH0	CSIH0 Configuration Register 2	CSIH0CFG2	0000 0000$_H$	32	FFD8 104C$_H$	2	32
CSIH0	CSIH0 Configuration Register 3	CSIH0CFG3	0000 0000$_H$	32	FFD8 1050$_H$	2	32
CSIH0	CSIH0 Configuration Register 4	CSIH0CFG4	0000 0000$_H$	32	FFD8 1054$_H$	2	32
CSIH0	CSIH0 Configuration Register 5	CSIH0CFG5	0000 0000$_H$	32	FFD8 1058$_H$	2	32
CSIH0	CSIH0 Baud Rate Setting Register 0	CSIH0BRS0	0000$_H$	16	FFD8 1068$_H$	2	16
CSIH0	CSIH0 Baud Rate Setting Register 1	CSIH0BRS1	0000$_H$	16	FFD8 106C$_H$	2	16
CSIH0	CSIH0 Baud Rate Setting Register 2	CSIH0BRS2	0000$_H$	16	FFD8 1070$_H$	2	16
CSIH0	CSIH0 Baud Rate Setting Register 3	CSIH0BRS3	0000$_H$	16	FFD8 1074$_H$	2	16
CSIH1	CSIH1 Control Register 0	CSIH1CTL0	00$_H$	8	FFD8 2000$_H$	2	8
CSIH1	CSIH1 Status Register 0	CSIH1STR0	0000 0010$_H$	32	FFD8 2004$_H$	2	32
CSIH1	CSIH1 Status Clear Register 0	CSIH1STCR0	0000$_H$	16	FFD8 2008$_H$	2	16
CSIH1	CSIH1 Control Register 1	CSIH1CTL1	0000 0000$_H$	32	FFD8 2010$_H$	2	32
CSIH1	CSIH1 Control Register 2	CSIH1CTL2	E000$_H$	16	FFD8 2014$_H$	2	16
CSIH1	CSIH1 Transmission Data Register 0 for Word Access	CSIH1TX0W	XXXX XXXX$_H$	32	FFD8 3008$_H$	2	32
CSIH1	CSIH1 Transmission Data Register 0 for Half Word Access	CSIH1TX0H	XXXX$_H$	16	FFD8 300C$_H$	2	16
CSIH1	CSIH1 Receive Data Register 0 for Word Access	CSIH1RX0W	XXXX XXXX$_H$	32	FFD8 3010$_H$	2	32
CSIH1	CSIH1 Receive Data Register 0 for Half Word Access	CSIH1RX0H	XXXX$_H$	16	FFD8 3014$_H$	2	16
CSIH1	CSIH1 Configuration Register 0	CSIH1CFG0	0000 0000$_H$	32	FFD8 3044$_H$	2	32
CSIH1	CSIH1 Configuration Register 1	CSIH1CFG1	0000 0000$_H$	32	FFD8 3048$_H$	2	32
CSIH1	CSIH1 Configuration Register 2	CSIH1CFG2	0000 0000$_H$	32	FFD8 304C$_H$	2	32
CSIH1	CSIH1 Configuration Register 3	CSIH1CFG3	0000 0000$_H$	32	FFD8 3050$_H$	2	32
CSIH1	CSIH1 Baud Rate Setting Register 0	CSIH1BRS0	0000$_H$	16	FFD8 3068$_H$	2	16

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
CSIH1	CSIH1 Baud Rate Setting Register 1	CSIH1BRS1	0000_H	16	FFD8 $306C_H$	2	16
CSIH1	CSIH1 Baud Rate Setting Register 2	CSIH1BRS2	0000_H	16	FFD8 3070_H	2	16
CSIH1	CSIH1 Baud Rate Setting Register 3	CSIH1BRS3	0000_H	16	FFD8 3074_H	2	16
CSIH2	CSIH2 Control Register 0	CSIH2CTL0	00_H	8	FFD8 4000_H	2	8
CSIH2	CSIH2 Status Register 0	CSIH2STR0	$0000\ 0010_H$	32	FFD8 4004_H	2	32
CSIH2	CSIH2 Status Clear Register 0	CSIH2STCR0	0000_H	16	FFD8 4008_H	2	16
CSIH2	CSIH2 Control Register 1	CSIH2CTL1	$0000\ 0000_H$	32	FFD8 4010_H	2	32
CSIH2	CSIH2 Control Register 2	CSIH2CTL2	$E000_H$	16	FFD8 4014_H	2	16
CSIH2	CSIH2 Word Access Transmission Data Register 0	CSIH2TX0W	$XXXX\ XXXX_H$	32	FFD8 5008_H	2	32
CSIH2	CSIH2 Half Word Access Transmission Data Register 0	CSIH2TX0H	$XXXX_H$	16	FFD8 $500C_H$	2	16
CSIH2	CSIH2 Word Access Receive Data Register 0	CSIH2RX0W	$XXXX\ XXXX_H$	32	FFD8 5010_H	2	32
CSIH2	CSIH2 Half Word Access Receive Data Register 0	CSIH2RX0H	$XXXX_H$	16	FFD8 5014_H	2	16
CSIH2	CSIH2 Configuration Register 0	CSIH2CFG0	$0000\ 0000_H$	32	FFD8 5044_H	2	32
CSIH2	CSIH2 Configuration Register 1	CSIH2CFG1	$0000\ 0000_H$	32	FFD8 5048_H	2	32
CSIH2	CSIH2 Configuration Register 2	CSIH2CFG2	$0000\ 0000_H$	32	FFD8 $504C_H$	2	32
CSIH2	CSIH2 Configuration Register 3	CSIH2CFG3	$0000\ 0000_H$	32	FFD8 5050_H	2	32
CSIH2	CSIH2 Baud Rate Setting Register 0	CSIH2BRS0	0000_H	16	FFD8 5068_H	2	16
CSIH2	CSIH2 Baud Rate Setting Register 1	CSIH2BRS1	0000_H	16	FFD8 $506C_H$	2	16
CSIH2	CSIH2 Baud Rate Setting Register 2	CSIH2BRS2	0000_H	16	FFD8 5070_H	2	16
CSIH2	CSIH2 Baud Rate Setting Register 3	CSIH2BRS3	0000_H	16	FFD8 5074_H	2	16
CSIH3	CSIH3 Control Register 0	CSIH3CTL0	00_H	8	FFD8 6000_H	2	8
CSIH3	CSIH3 Status Register 0	CSIH3STR0	$0000\ 0010_H$	32	FFD8 6004_H	2	32
CSIH3	CSIH3 Status Clear Register 0	CSIH3STCR0	0000_H	16	FFD8 6008_H	2	16
CSIH3	CSIH3 Control Register 1	CSIH3CTL1	$0000\ 0000_H$	32	FFD8 6010_H	2	32
CSIH3	CSIH3 Control Register 2	CSIH3CTL2	$E000_H$	16	FFD8 6014_H	2	16
CSIH3	CSIH3 Word Access Transmission Data Register 0	CSIH3TX0W	$XXXX\ XXXX_H$	32	FFD8 7008_H	2	32
CSIH3	CSIH3 Transmission Data Register 0 for Half Word Access	CSIH3TX0H	$XXXX_H$	16	FFD8 $700C_H$	2	16
CSIH3	CSIH3 Receive Data Register 0 for Word Access	CSIH3RX0W	$XXXX\ XXXX_H$	32	FFD8 7010_H	2	32
CSIH3	CSIH3 Receive Data Register 0 for Half Word Access	CSIH3RX0H	$XXXX_H$	16	FFD8 7014_H	2	16
CSIH3	CSIH3 Configuration Register 0	CSIH3CFG0	$0000\ 0000_H$	32	FFD8 7044_H	2	32
CSIH3	CSIH3 Configuration Register 1	CSIH3CFG1	$0000\ 0000_H$	32	FFD8 7048_H	2	32
CSIH3	CSIH3 Configuration Register 2	CSIH3CFG2	$0000\ 0000_H$	32	FFD8 $704C_H$	2	32
CSIH3	CSIH3 Configuration Register 3	CSIH3CFG3	$0000\ 0000_H$	32	FFD8 7050_H	2	32
CSIH3	CSIH3 Baud Rate Setting Register 0	CSIH3BRS0	0000_H	16	FFD8 7068_H	2	16
CSIH3	CSIH3 Baud Rate Setting Register 1	CSIH3BRS1	0000_H	16	FFD8 $706C_H$	2	16
CSIH3	CSIH3 Baud Rate Setting Register 2	CSIH3BRS2	0000_H	16	FFD8 7070_H	2	16
CSIH3	CSIH3 Baud Rate Setting Register 3	CSIH3BRS3	0000_H	16	FFD8 7074_H	2	16
SCI30	Serial Mode Register	SCI30SMR	00_H	8	FFD9 0000_H	1(1L)	8
SCI30	Bit Rate Register	SCI30BRR	FF_H	8	FFD9 0004_H	1(1L)	8
SCI30	Modulation Duty Register	SCI30MDDR	FF_H	8	FFD9 0004_H	1(1L)	8
SCI30	Serial Control Register	SCI30SCR	00_H	8	FFD9 0008_H	1(1L)	8
SCI30	Transmission Data Register	SCI30TDR	FF_H	8	FFD9 $000C_H$	1(1L)	8

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
SCI30	Serial Status Register	SCI30SSR	84_H	8	FFD9 0010_H	1(1L)	8
SCI30	Receive Data Register	SCI30RDR	00_H	8	FFD9 0014_H	1(1L)	8
SCI30	Serial Communication Format Register	SCI30SCMR	$F2_H$	8	FFD9 0018_H	1(1L)	8
SCI30	Serial Expansion Mode Register	SCI30SEMR	04_H	8	FFD9 $001C_H$	1(1L)	8
SCI31	Serial Mode Register	SCI31SMR	00_H	8	FFD9 1000_H	1(1L)	8
SCI31	Bit Rate Register	SCI31BRR	FF_H	8	FFD9 1004_H	1(1L)	8
SCI31	Modulation Duty Register	SCI31MDDR	FF_H	8	FFD9 1004_H	1(1L)	8
SCI31	Serial Control Register	SCI31SCR	00_H	8	FFD9 1008_H	1(1L)	8
SCI31	Transmission Data Register	SCI31TDR	FF_H	8	FFD9 $100C_H$	1(1L)	8
SCI31	Serial Status Register	SCI31SSR	84_H	8	FFD9 1010_H	1(1L)	8
SCI31	Receive Data Register	SCI31RDR	00_H	8	FFD9 1014_H	1(1L)	8
SCI31	Serial Communication Format Register	SCI31SCMR	$F2_H$	8	FFD9 1018_H	1(1L)	8
SCI31	Serial Expansion Mode Register	SCI31SEMR	04_H	8	FFD9 $101C_H$	1(1L)	8
SCI32	Serial Mode Register	SCI32SMR	00_H	8	FFD9 2000_H	1(1L)	8
SCI32	Bit Rate Register	SCI32BRR	FF_H	8	FFD9 2004_H	1(1L)	8
SCI32	Modulation Duty Register	SCI32MDDR	FF_H	8	FFD9 2004_H	1(1L)	8
SCI32	Serial Control Register	SCI32SCR	00_H	8	FFD9 2008_H	1(1L)	8
SCI32	Transmission Data Register	SCI32TDR	FF_H	8	FFD9 $200C_H$	1(1L)	8
SCI32	Serial Status Register	SCI32SSR	84_H	8	FFD9 2010_H	1(1L)	8
SCI32	Receive Data Register	SCI32RDR	00_H	8	FFD9 2014_H	1(1L)	8
SCI32	Serial Communication Format Register	SCI32SCMR	$F2_H$	8	FFD9 2018_H	1(1L)	8
SCI32	Serial Expansion Mode Register	SCI32SEMR	04_H	8	FFD9 $201C_H$	1(1L)	8
SCI33	Serial Mode Register	SCI33SMR	00_H	8	FFD9 3000_H	1(1L)	8
SCI33	Bit Rate Register	SCI33BRR	FF_H	8	FFD9 3004_H	1(1L)	8
SCI33	Modulation Duty Register	SCI33MDDR	FF_H	8	FFD9 3004_H	1(1L)	8
SCI33	Serial Control Register	SCI33SCR	00_H	8	FFD9 3008_H	1(1L)	8
SCI33	Transmission Data Register	SCI33TDR	FF_H	8	FFD9 $300C_H$	1(1L)	8
SCI33	Serial Status Register	SCI33SSR	84_H	8	FFD9 3010_H	1(1L)	8
SCI33	Receive Data Register	SCI33RDR	00_H	8	FFD9 3014_H	1(1L)	8
SCI33	Serial Communication Format Register	SCI33SCMR	$F2_H$	8	FFD9 3018_H	1(1L)	8
SCI33	Serial Expansion Mode Register	SCI33SEMR	04_H	8	FFD9 $301C_H$	1(1L)	8
PBG	PBG1A Protection Register 0	FSGD1ADPROT0	07FF $FFFF_H$	32	FFDC 0000_H	1(1L)	8, 16, 32
PBG	PBG1A Protection Register 1	FSGD1ADPROT1	07FF $FFFF_H$	32	FFDC 0004_H	1(1L)	8, 16, 32
PBG	PBG1A Error Control Register	ERRSLV1ACTL	0000 0000_H	32	FFDC 0200_H	1(1L)	8, 16, 32
PBG	PBG1A Error Status Register	ERRSLV1ASTAT	0000 0000_H	32	FFDC 0204_H	1(1L)	8, 16, 32
PBG	PBG1A Error Address Register	ERRSLV1AADDR	0000 0000_H	32	FFDC 0208_H	1(1L)	32
PBG	PBG1A Error Type Register	ERRSLV1ATYPE	0000 0000_H	32	FFDC $020C_H$	1(1L)	16, 32
APDP	P-Bus Data Parity Status Register APAA	APDPERRST_APAA	0000 0000_H	32	FFDC 2000_H	1(1L)	8, 16, 32
APDP	P-Bus Data Parity Status Clear Register APAA	APDPERRSTC_APAA	0000 0000_H	32	FFDC 2004_H	1(1L)	8, 16, 32
APDP	P-Bus Data Parity Test Mode Control Register APAA	APDPTMC_APAA	0000 0000_H	32	FFDC 2008_H	1(1L)	16, 32
APDP	P-Bus Data Parity Error Address Register APAA	APDPERRADR_APAA	0000 0000_H	32	FFDC $200C_H$	1(1L)	32
APDP	P-Bus Data Parity Status Register APGA	APDPERRST_APGA	0000 0000_H	32	FFDC 2020_H	1(1L)	8, 16, 32
APDP	P-Bus Data Parity Status Clear Register APGA	APDPERRSTC_APGA	0000 0000_H	32	FFDC 2024_H	1(1L)	8, 16, 32
APDP	P-Bus Data Parity Test Mode Control Register APGA	APDPTMC_APGA	0000 0000_H	32	FFDC 2028_H	1(1L)	16, 32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
APDP	P-Bus Data Parity Error Address Register APGA	APDPERRADR_APGA	0000 0000$_H$	32	FFDC 202C$_H$	1(1L)	32
PIC1	Control Register EN	PIC1EN	00$_H$	8	FFDD 0000$_H$	1(1H)	8
PIC1	Simultaneous Start Trigger Control Register	PIC1SST	00$_H$	8	FFDD 0004$_H$	1(1H)	8
PIC1	Simultaneous Start Control Register 2	PIC1SSER2	0000$_H$	16	FFDD 0018$_H$	1(1H)	16
PIC1	Simultaneous Start Control Register 3	PIC1SSER3	0000$_H$	16	FFDD 001C$_H$	1(1H)	16
PIC1	Hi-Z Output Control Register 2	PIC1HIZCEN2	00$_H$	8	FFDD 0088$_H$	1(1H)	8
PIC1	Hi-Z Output Control Register 3	PIC1HIZCEN3	00$_H$	8	FFDD 008C$_H$	1(1H)	8
PIC2	AD Converter 0 Trigger Selection Control Register 0	PIC2ADCB0TSEL0	0000 0000$_H$	32	FFDD 1000$_H$	1(1H)	16, 32
PIC2	AD Converter 0 Trigger Selection Control Register 1	PIC2ADCB0TSEL1	0000 0000$_H$	32	FFDD 1004$_H$	1(1H)	16, 32
PIC2	AD Converter 0 Trigger Selection Control Register 2	PIC2ADCB0TSEL2	0000 0000$_H$	32	FFDD 1008$_H$	1(1H)	16, 32
PIC2	AD Converter 0 Trigger Selection Control Register 3	PIC2ADCB0TSEL3	0000 0000$_H$	32	FFDD 100C$_H$	1(1H)	16, 32
PIC2	AD Converter 0 Trigger Selection Control Register 4	PIC2ADCB0TSEL4	0000 0000$_H$	32	FFDD 1010$_H$	1(1H)	16, 32
PIC2	AD Converter 0 Trigger Edge Selection Control Register	PIC2ADCB0EDGSEL	0000$_H$	16	FFDD 101C$_H$	1(1H)	16
PIC2	AD Converter 1 Trigger Selection Control Register 0	PIC2ADCB1TSEL0	0000 0000$_H$	32	FFDD 1020$_H$	1(1H)	16, 32
PIC2	AD Converter 1 Trigger Selection Control Register 1	PIC2ADCB1TSEL1	0000 0000$_H$	32	FFDD 1024$_H$	1(1H)	16, 32
PIC2	AD Converter 1 Trigger Selection Control Register 2	PIC2ADCB1TSEL2	0000 0000$_H$	32	FFDD 1028$_H$	1(1H)	16, 32
PIC2	AD Converter 1 Trigger Selection Control Register 3	PIC2ADCB1TSEL3	0000 0000$_H$	32	FFDD 102C$_H$	1(1H)	16, 32
PIC2	AD Converter 1 Trigger Selection Control Register 4	PIC2ADCB1TSEL4	0000 0000$_H$	32	FFDD 1030$_H$	1(1H)	16, 32
PIC2	AD Converter 1 Trigger Edge Selection Control Register	PIC2ADCB1EDGSEL	0000$_H$	16	FFDD 103C$_H$	1(1H)	16
PIC2	AD Converter Trigger Output Control Register 500	PIC2ADTEN500	0000 0000$_H$	32	FFDD 1080$_H$	1(1H)	16, 32
PIC2	AD Converter Trigger Output Control Register 501	PIC2ADTEN501	0000 0000$_H$	32	FFDD 1084$_H$	1(1H)	16, 32
PIC2	AD Converter Trigger Output Control Register 502	PIC2ADTEN502	0000 0000$_H$	32	FFDD 1088$_H$	1(1H)	16, 32
PIC2	AD Converter Trigger Output Control Register 503	PIC2ADTEN503	0000 0000$_H$	32	FFDD 108C$_H$	1(1H)	16, 32
PIC2	AD Converter Trigger Output Control Register 504	PIC2ADTEN504	0000 0000$_H$	32	FFDD 1090$_H$	1(1H)	16, 32
PIC2	AD Converter Trigger Output Control Register 510	PIC2ADTEN510	0000 0000$_H$	32	FFDD 1094$_H$	1(1H)	16, 32
PIC2	AD Converter Trigger Output Control Register 511	PIC2ADTEN511	0000 0000$_H$	32	FFDD 1098$_H$	1(1H)	16, 32
PIC2	AD Converter Trigger Output Control Register 512	PIC2ADTEN512	0000 0000$_H$	32	FFDD 109C$_H$	1(1H)	16, 32
PIC2	AD Converter Trigger Output Control Register 513	PIC2ADTEN513	0000 0000$_H$	32	FFDD 10A0$_H$	1(1H)	16, 32
PIC2	AD Converter Trigger Output Control Register 514	PIC2ADTEN514	0000 0000$_H$	32	FFDD 10A4$_H$	1(1H)	16, 32
PIC2	AD Converter Trigger Output Control Register 600	PIC2ADTEN600	0000 0000$_H$	32	FFDD 10A8$_H$	1(1H)	32
PIC2	AD Converter Trigger Output Control Register 601	PIC2ADTEN601	0000 0000$_H$	32	FFDD 10AC$_H$	1(1H)	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
PIC2	AD Converter Trigger Output Control Register 602	PIC2ADTEN602	0000 0000$_H$	32	FFDD 10B0$_H$	1(1H)	32
PIC2	AD Converter Trigger Output Control Register 603	PIC2ADTEN603	0000 0000$_H$	32	FFDD 10B4$_H$	1(1H)	32
PIC2	AD Converter Trigger Output Control Register 604	PIC2ADTEN604	0000 0000$_H$	32	FFDD 10B8$_H$	1(1H)	32
PIC2	AD Converter Trigger Output Control Register 610	PIC2ADTEN610	0000 0000$_H$	32	FFDD 10BC$_H$	1(1H)	32
PIC2	AD Converter Trigger Output Control Register 611	PIC2ADTEN611	0000 0000$_H$	32	FFDD 10C0$_H$	1(1H)	32
PIC2	AD Converter Trigger Output Control Register 612	PIC2ADTEN612	0000 0000$_H$	32	FFDD 10C4$_H$	1(1H)	32
PIC2	AD Converter Trigger Output Control Register 613	PIC2ADTEN613	0000 0000$_H$	32	FFDD 10C8$_H$	1(1H)	32
PIC2	AD Converter Trigger Output Control Register 614	PIC2ADTEN614	0000 0000$_H$	32	FFDD 10CC$_H$	1(1H)	32
PIC2	AD Converter Trigger Output Control Register 700	PIC2ADTEN700	0000 0000$_H$	32	FFDD 10D0$_H$	1(1H)	32
PIC2	AD Converter Trigger Output Control Register 701	PIC2ADTEN701	0000 0000$_H$	32	FFDD 10D4$_H$	1(1H)	32
PIC2	AD Converter Trigger Output Control Register 702	PIC2ADTEN702	0000 0000$_H$	32	FFDD 10D8$_H$	1(1H)	32
PIC2	AD Converter Trigger Output Control Register 703	PIC2ADTEN703	0000 0000$_H$	32	FFDD 10DC$_H$	1(1H)	32
PIC2	AD Converter Trigger Output Control Register 704	PIC2ADTEN704	0000 0000$_H$	32	FFDD 10E0$_H$	1(1H)	32
PIC2	AD Converter Trigger Output Control Register 710	PIC2ADTEN710	0000 0000$_H$	32	FFDD 10E4$_H$	1(1H)	32
PIC2	AD Converter Trigger Output Control Register 711	PIC2ADTEN711	0000 0000$_H$	32	FFDD 10E8$_H$	1(1H)	32
PIC2	AD Converter Trigger Output Control Register 712	PIC2ADTEN712	0000 0000$_H$	32	FFDD 10EC$_H$	1(1H)	32
PIC2	AD Converter Trigger Output Control Register 713	PIC2ADTEN713	0000 0000$_H$	32	FFDD 10F0$_H$	1(1H)	32
PIC2	AD Converter Trigger Output Control Register 714	PIC2ADTEN714	0000 0000$_H$	32	FFDD 10F4$_H$	1(1H)	32
PIC2	DSADC Start Trigger Output Control Register 00	PIC2DSADTEN000	0000$_H$	16	FFDD 1100$_H$	1(1H)	16
PIC2	DSADC Start Trigger Output Control Register 01	PIC2DSADTEN001	0000 0000$_H$	32	FFDD 1104$_H$	1(1H)	32
PIC2	DSADC Start Trigger Output Control Register 02	PIC2DSADTEN002	0000 0000$_H$	32	FFDD 1108$_H$	1(1H)	32
PIC2	DSADC Stop Trigger Output Control Register 00	PIC2DSADTEN100	0000$_H$	16	FFDD 110C$_H$	1(1H)	16
PIC2	DSADC Stop Trigger Output Control Register 01	PIC2DSADTEN101	0000 0000$_H$	32	FFDD 1110$_H$	1(1H)	32
PIC2	DSADC Stop Trigger Output Control Register 02	PIC2DSADTEN102	0000 0000$_H$	32	FFDD 1114$_H$	1(1H)	32
PIC2	DSADC Start Trigger Output Control Register 10	PIC2DSADTEN010	0000$_H$	16	FFDD 1118$_H$	1(1H)	16
PIC2	DSADC Start Trigger Output Control Register 11	PIC2DSADTEN011	0000 0000$_H$	32	FFDD 111C$_H$	1(1H)	32
PIC2	DSADC Start Trigger Output Control Register 12	PIC2DSADTEN012	0000 0000$_H$	32	FFDD 1120$_H$	1(1H)	32
PIC2	DSADC Stop Trigger Output Control Register 10	PIC2DSADTEN110	0000$_H$	16	FFDD 1124$_H$	1(1H)	16
PIC2	DSADC Stop Trigger Output Control Register 11	PIC2DSADTEN111	0000 0000$_H$	32	FFDD 1128$_H$	1(1H)	32
PIC2	DSADC Stop Trigger Output Control Register 12	PIC2DSADTEN112	0000 0000$_H$	32	FFDD 112C$_H$	1(1H)	32

(67/201)

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
PIC2	DSADC Start Trigger Output Control Register 20	PIC2DSADTEN020	0000_H	16	FFDD 1130_H	1(1H)	16
PIC2	DSADC Start Trigger Output Control Register 21	PIC2DSADTEN021	$0000\ 0000_H$	32	FFDD 1134_H	1(1H)	32
PIC2	DSADC Start Trigger Output Control Register 22	PIC2DSADTEN022	$0000\ 0000_H$	32	FFDD 1138_H	1(1H)	32
PIC2	DSADC Stop Trigger Output Control Register 20	PIC2DSADTEN120	0000_H	16	FFDD $113C_H$	1(1H)	16
PIC2	DSADC Stop Trigger Output Control Register 21	PIC2DSADTEN121	$0000\ 0000_H$	32	FFDD 1140_H	1(1H)	32
PIC2	DSADC Stop Trigger Output Control Register 22	PIC2DSADTEN122	$0000\ 0000_H$	32	FFDD 1144_H	1(1H)	32
PIC2	DSADC Start Trigger Output Control Register 30	PIC2DSADTEN030	0000_H	16	FFDD 1148_H	1(1H)	16
PIC2	DSADC Start Trigger Output Control Register 31	PIC2DSADTEN031	$0000\ 0000_H$	32	FFDD $114C_H$	1(1H)	32
PIC2	DSADC Start Trigger Output Control Register 32	PIC2DSADTEN032	$0000\ 0000_H$	32	FFDD 1150_H	1(1H)	32
PIC2	DSADC Stop Trigger Output Control Register 30	PIC2DSADTEN130	0000_H	16	FFDD 1154_H	1(1H)	16
PIC2	DSADC Stop Trigger Output Control Register 31	PIC2DSADTEN131	$0000\ 0000_H$	32	FFDD 1158_H	1(1H)	32
PIC2	DSADC Stop Trigger Output Control Register 32	PIC2DSADTEN132	$0000\ 0000_H$	32	FFDD $115C_H$	1(1H)	32
PIC2	DSADC Start Trigger Output Control Register 40	PIC2DSADTEN040	0000_H	16	FFDD 1160_H	1(1H)	16
PIC2	DSADC Start Trigger Output Control Register 41	PIC2DSADTEN041	$0000\ 0000_H$	32	FFDD 1164_H	1(1H)	32
PIC2	DSADC Start Trigger Output Control Register 42	PIC2DSADTEN042	$0000\ 0000_H$	32	FFDD 1168_H	1(1H)	32
PIC2	DSADC Stop Trigger Output Control Register 40	PIC2DSADTEN140	0000_H	16	FFDD $116C_H$	1(1H)	16
PIC2	DSADC Stop Trigger Output Control Register 41	PIC2DSADTEN141	$0000\ 0000_H$	32	FFDD 1170_H	1(1H)	32
PIC2	DSADC Stop Trigger Output Control Register 42	PIC2DSADTEN142	$0000\ 0000_H$	32	FFDD 1174_H	1(1H)	32
PIC2	DSADC Start Trigger Output Control Register 50	PIC2DSADTEN050	0000_H	16	FFDD 1178_H	1(1H)	16
PIC2	DSADC Start Trigger Output Control Register 51	PIC2DSADTEN051	$0000\ 0000_H$	32	FFDD $117C_H$	1(1H)	32
PIC2	DSADC Start Trigger Output Control Register 52	PIC2DSADTEN052	$0000\ 0000_H$	32	FFDD 1180_H	1(1H)	32
PIC2	DSADC Stop Trigger Output Control Register 50	PIC2DSADTEN150	0000_H	16	FFDD 1184_H	1(1H)	16
PIC2	DSADC Stop Trigger Output Control Register 51	PIC2DSADTEN151	$0000\ 0000_H$	32	FFDD 1188_H	1(1H)	32
PIC2	DSADC Stop Trigger Output Control Register 52	PIC2DSADTEN152	$0000\ 0000_H$	32	FFDD $118C_H$	1(1H)	32
PIC2	DSADC Start Trigger Output Control Register 60	PIC2DSADTEN060	0000_H	16	FFDD 1190_H	1(1H)	16
PIC2	DSADC Start Trigger Output Control Register 61	PIC2DSADTEN061	$0000\ 0000_H$	32	FFDD 1194_H	1(1H)	32
PIC2	DSADC Start Trigger Output Control Register 62	PIC2DSADTEN062	$0000\ 0000_H$	32	FFDD 1198_H	1(1H)	32
PIC2	DSADC Stop Trigger Output Control Register 60	PIC2DSADTEN160	0000_H	16	FFDD $119C_H$	1(1H)	16
PIC2	DSADC Stop Trigger Output Control Register 61	PIC2DSADTEN161	$0000\ 0000_H$	32	FFDD $11A0_H$	1(1H)	32
PIC2	DSADC Stop Trigger Output Control Register 62	PIC2DSADTEN162	$0000\ 0000_H$	32	FFDD $11A4_H$	1(1H)	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
PIC2	DSADC Start Trigger Output Control Register 70	PIC2DSADTEN070	0000_H	16	FFDD $11A8_H$	1(1H)	16
PIC2	DSADC Start Trigger Output Control Register 71	PIC2DSADTEN071	$0000\ 0000_H$	32	FFDD $11AC_H$	1(1H)	32
PIC2	DSADC Start Trigger Output Control Register 72	PIC2DSADTEN072	$0000\ 0000_H$	32	FFDD $11B0_H$	1(1H)	32
PIC2	DSADC Stop Trigger Output Control Register 70	PIC2DSADTEN170	0000_H	16	FFDD $11B4_H$	1(1H)	16
PIC2	DSADC Stop Trigger Output Control Register 71	PIC2DSADTEN171	$0000\ 0000_H$	32	FFDD $11B8_H$	1(1H)	32
PIC2	DSADC Stop Trigger Output Control Register 72	PIC2DSADTEN172	$0000\ 0000_H$	32	FFDD $11BC_H$	1(1H)	32
PIC2	DSADC Trigger Selection Control Register 0	PIC2DSADCATSEL0	0000_H	16	FFDD $11C0_H$	1(1H)	16
PIC2	DSADC Trigger Selection Control Register 1	PIC2DSADCATSEL1	0000_H	16	FFDD $11C4_H$	1(1H)	16
PBG	PBG1B Protection Register 0	FSGD1BDPROT0	$07FF\ FFFF_H$	32	FFDD $D000_H$	1(1H)	8, 16, 32
PBG	PBG1B Error Control Register	ERRSLV1BCTL	$0000\ 0000_H$	32	FFDD $D200_H$	1(1H)	8, 16, 32
PBG	PBG1B Error Status Register	ERRSLV1BSTAT	$0000\ 0000_H$	32	FFDD $D204_H$	1(1H)	8, 16, 32
PBG	PBG1B Error Address Register	ERRSLV1BADDR	$0000\ 0000_H$	32	FFDD $D208_H$	1(1H)	32
PBG	PBG1B Error Type Register	ERRSLV1BTYPE	$0000\ 0000_H$	32	FFDD $D20C_H$	1(1H)	16, 32
APDP	P-Bus Data Parity Status Register ATU4	APDPERRST_ATU4	$0000\ 0000_H$	32	FFDD $F000_H$	1(1H)	8, 16, 32
APDP	P-Bus Data Parity Status Clear Register ATU4	APDPERRSTC_ATU4	$0000\ 0000_H$	32	FFDD $F004_H$	1(1H)	8, 16, 32
APDP	P-Bus Data Parity Test Mode Control Register ATU4	APDPTMC_ATU4	$0000\ 0000_H$	32	FFDD $F008_H$	1(1H)	16, 32
APDP	P-Bus Data Parity Error Address Register ATU4	APDPERRADR_ATU4	$0000\ 0000_H$	32	FFDD $F00C_H$	1(1H)	32
ATU4CTRL	ATU-IV Master Enable Register	ATUENR	0000_H	16	FFE6 0000_H	1(1H)	8, 16
ATU4CTRL	Clock Bus Control Register	CBCNT	00_H	8	FFE6 0002_H	1(1H)	8
ATU4CTRL	Noise Cancel Mode Register	NCMR	00_H	8	FFE6 0003_H	1(1H)	8
ATU4CTRL	Prescaler Register 0	PSCR0	0000_H	16	FFE6 0080_H	1(1H)	16
ATU4CTRL	Prescaler Register 1	PSCR1	0000_H	16	FFE6 0084_H	1(1H)	16
ATU4CTRL	Prescaler Register 2	PSCR2	0000_H	16	FFE6 0088_H	1(1H)	16
ATU4CTRL	Prescaler Register 3	PSCR3	0000_H	16	FFE6 $008C_H$	1(1H)	16
ATU4CTRL	Trigger Status Register DMA0	TRGSRDMA0	00_H	8	FFE6 $00C0_H$	1(1H)	8
ATU4CTRL	Trigger Selection Register DMA00	TRGSELDMA00	00_H	8	FFE6 $00C4_H$	1(1H)	8
ATU4CTRL	Trigger Selection Register DMA01	TRGSELDMA01	00_H	8	FFE6 $00C6_H$	1(1H)	8
ATU4CTRL	Trigger Selection Register AD	TRGSELAD	00_H	8	FFE6 $00C8_H$	1(1H)	8
ATU4CTRL	Trigger Status Register DMA1	TRGSRDMA1	00_H	8	FFE6 $00D0_H$	1(1H)	8
ATU4CTRL	Trigger Selection Register DMA10	TRGSELDMA10	00_H	8	FFE6 $00D4_H$	1(1H)	8
ATU4CTRL	Trigger Selection Register DMA11	TRGSELDMA11	00_H	8	FFE6 $00D6_H$	1(1H)	8
ATU4H	Timer Control Register H	TCRH	00_H	8	FFE6 0100_H	1(1H)	8
ATU4H	Timer Status Register H	TSRH	00_H	8	FFE6 0102_H	1(1H)	8
ATU4H	Timer Status Clear Register H	TSCRH	00_H	8	FFE6 0103_H	1(1H)	8
ATU4H	Timer counter 1H	TCNT1H	0000_H	16	FFE6 0104_H	1(1H)	16
ATU4H	Compare match Register 1H	OCR1H	$FFFF_H$	16	FFE6 0106_H	1(1H)	16
ATU4H	Timer counter 2H	TCNT2H	$0000\ 0000_H$	32	FFE6 0108_H	1(1H)	32
ATU4A	Timer Control Register 1A	TCR1A	00_H	8	FFE6 0200_H	1(1H)	8
ATU4A	Timer Control Register 2A	TCR2A	00_H	8	FFE6 0202_H	1(1H)	8
ATU4A	Timer Control Register 3A	TCR3A	00_H	8	FFE6 0204_H	1(1H)	8

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
ATU4A	Timer Control Register 4A	TCR4A	00_H	8	FFE6 0206_H	1(1H)	8
ATU4A	Timer Status Register A	TSRA	00_H	8	FFE6 0208_H	1(1H)	8
ATU4A	Timer Status Clear Register A	TSCRA	00_H	8	FFE6 $020A_H$	1(1H)	8
ATU4A	Noise Cancel Mode Channel Register 1A	NCMCR1A	00_H	8	FFE6 $020C_H$	1(1H)	8
ATU4A	Noise Cancel Mode Channel Register 2A	NCMCR2A	00_H	8	FFE6 $020E_H$	1(1H)	8
ATU4A	Timer I/O Control Register 1A	TIOR1A	0000_H	16	FFE6 0210_H	1(1H)	8, 16
ATU4A	Timer I/O Control Register 2A	TIOR2A	$0000\ 0000_H$	32	FFE6 0214_H	1(1H)	8, 16, 32
ATU4A	Timer Input Signal Level Register A	TILRA	00_H	8	FFE6 0218_H	1(1H)	8
ATU4A	Timer Input Signal Level Capture Register A	TILCRA	00_H	8	FFE6 $021A_H$	1(1H)	8
ATU4A	Free-Running Counter A	TCNTA	$0000\ 0000_H$	32	FFE6 0220_H	1(1H)	32
ATU4A	Input Capture Register A0	ICRA0	$0000\ 0000_H$	32	FFE6 0240_H	1(1H)	32
ATU4A	Noise Cancel Counter A0	NCNTA0	0000_H	16	FFE6 0244_H	1(1H)	16
ATU4A	Noise cancel Register A0	NCRA0	0000_H	16	FFE6 0246_H	1(1H)	16
ATU4A	Input Capture Register A1	ICRA1	$0000\ 0000_H$	32	FFE6 0260_H	1(1H)	32
ATU4A	Noise Cancel Counter A1	NCNTA1	0000_H	16	FFE6 0264_H	1(1H)	16
ATU4A	Noise cancel Register A1	NCRA1	0000_H	16	FFE6 0266_H	1(1H)	16
ATU4A	Input Capture Register A2	ICRA2	$0000\ 0000_H$	32	FFE6 0280_H	1(1H)	32
ATU4A	Noise Cancel Counter A2	NCNTA2	0000_H	16	FFE6 0284_H	1(1H)	16
ATU4A	Noise cancel Register A2	NCRA2	0000_H	16	FFE6 0286_H	1(1H)	16
ATU4A	Input Capture Register A3	ICRA3	$0000\ 0000_H$	32	FFE6 $02A0_H$	1(1H)	32
ATU4A	Noise Cancel Counter A3	NCNTA3	0000_H	16	FFE6 $02A4_H$	1(1H)	16
ATU4A	Noise cancel Register A3	NCRA3	0000_H	16	FFE6 $02A6_H$	1(1H)	16
ATU4A	Input Capture Register A4	ICRA4	$0000\ 0000_H$	32	FFE6 $02C0_H$	1(1H)	32
ATU4A	Noise Cancel Counter A4	NCNTA4	0000_H	16	FFE6 $02C4_H$	1(1H)	16
ATU4A	Noise cancel Register A4	NCRA4	0000_H	16	FFE6 $02C6_H$	1(1H)	16
ATU4A	Input Capture Register A5	ICRA5	$0000\ 0000_H$	32	FFE6 $02E0_H$	1(1H)	32
ATU4A	Noise Cancel Counter A5	NCNTA5	0000_H	16	FFE6 $02E4_H$	1(1H)	16
ATU4A	Noise cancel Register A5	NCRA5	0000_H	16	FFE6 $02E6_H$	1(1H)	16
ATU4A	Input Capture Register A6	ICRA6	$0000\ 0000_H$	32	FFE6 0300_H	1(1H)	32
ATU4A	Noise Cancel Counter A6	NCNTA6	0000_H	16	FFE6 0304_H	1(1H)	16
ATU4A	Noise cancel Register A6	NCRA6	0000_H	16	FFE6 0306_H	1(1H)	16
ATU4B	Timer Control Register B	TCRB	00_H	8	FFE6 0400_H	1(1H)	8
ATU4B	Timer I/O Control Register B	TIORB	00_H	8	FFE6 0402_H	1(1H)	8
ATU4B	Timer Status Register B	TSRB	0000_H	16	FFE6 0404_H	1(1H)	8, 16
ATU4B	Timer Status Clear Register B	TSCRB	0000_H	16	FFE6 0406_H	1(1H)	8, 16
ATU4B	Timer Interrupt Control Register B	TICRB	00_H	8	FFE6 0408_H	1(1H)	8
ATU4B	Edge Interval Measuring Counter B0	TCNTB0	$0000\ 0001_H$	32	FFE6 0480_H	1(1H)	32
ATU4B	Input Capture Register B0	ICRB0	$0000\ 0000_H$	32	FFE6 0484_H	1(1H)	32
ATU4B	Record Register B1	RECRB1	$0000\ 0000_H$	32	FFE6 0488_H	1(1H)	32
ATU4B	Record Register B2	RECRB2	$0000\ 0000_H$	32	FFE6 $048C_H$	1(1H)	32
ATU4B	Record Register B3	RECRB3	$0000\ 0000_H$	32	FFE6 0490_H	1(1H)	32
ATU4B	Record Register B4	RECRB4	$0000\ 0000_H$	32	FFE6 0494_H	1(1H)	32
ATU4B	Record Register B5	RECRB5	$0000\ 0000_H$	32	FFE6 0498_H	1(1H)	32
ATU4B	Record Register B6	RECRB6	$0000\ 0000_H$	32	FFE6 $049C_H$	1(1H)	32
ATU4B	Record Back-up Register B0	RBURB0	$0000\ 0000_H$	32	FFE6 $04A0_H$	1(1H)	32

(70/201)

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
ATU4B	Record Back-up Register B1	RBURB1	0000 0000_H	32	FFE6 $04A4_H$	1(1H)	32
ATU4B	Record Back-up Register B2	RBURB2	0000 0000_H	32	FFE6 $04A8_H$	1(1H)	32
ATU4B	Record Back-up Register B3	RBURB3	0000 0000_H	32	FFE6 $04AC_H$	1(1H)	32
ATU4B	Record Back-up Register B4	RBURB4	0000 0000_H	32	FFE6 $04B0_H$	1(1H)	32
ATU4B	Record Back-up Register B5	RBURB5	0000 0000_H	32	FFE6 $04B4_H$	1(1H)	32
ATU4B	Record Back-up Register B6	RBURB6	0000 0000_H	32	FFE6 $04B8_H$	1(1H)	32
ATU4B	Output Comparison Register B0	OCRB0	FFFF $FFFF_H$	32	FFE6 $04BC_H$	1(1H)	32
ATU4B	Event Counter B1	TCNTB1	00_H	8	FFE6 $04C4_H$	1(1H)	8
ATU4B	Output Comparison Register B1	OCRB1	FF_H	8	FFE6 $04C6_H$	1(1H)	8
ATU4B	Output Comparison Register B10	OCRB10	FF_H	8	FFE6 $04C8_H$	1(1H)	8
ATU4B	Output Comparison Register B11	OCRB11	FF_H	8	FFE6 $04CA_H$	1(1H)	8
ATU4B	Output Comparison Register B12	OCRB12	FF_H	8	FFE6 $04CC_H$	1(1H)	8
ATU4B	Input Capture Register B1	ICRB1	0000 0000_H	32	FFE6 $04D0_H$	1(1H)	32
ATU4B	Input Capture Register B2	ICRB2	0000 0000_H	32	FFE6 $04D4_H$	1(1H)	32
ATU4B	Input Capture Register B30	ICRB30	00_H	8	FFE6 $04D8_H$	1(1H)	8
ATU4B	Input Capture Register B31	ICRB31	00_H	8	FFE6 $04D9_H$	1(1H)	8
ATU4B	Input Capture Register B32	ICRB32	00_H	8	FFE6 $04DA_H$	1(1H)	8
ATU4B	Input Capture Register B33	ICRB33	00_H	8	FFE6 $04DB_H$	1(1H)	8
ATU4B	Input Capture Register B34	ICRB34	00_H	8	FFE6 $04DC_H$	1(1H)	8
ATU4B	Input Capture Register B35	ICRB35	00_H	8	FFE6 $04DD_H$	1(1H)	8
ATU4B	Input Capture Register B36	ICRB36	00_H	8	FFE6 $04DE_H$	1(1H)	8
ATU4B	Load Register B	LDB	0000 0000_H	32	FFE6 0500_H	1(1H)	32
ATU4B	Reload Register B	RLDB	0000 0000_H	32	FFE6 0504_H	1(1H)	32
ATU4B	Reload Counter B2	TCNTB2	0000 0000_H	32	FFE6 0508_H	1(1H)	32
ATU4B	Pulse Interval Multiplier Register 1	PIMR1	0001_H	16	FFE6 $050C_H$	1(1H)	16
ATU4B	Pulse Interval Multiplier Register 2	PIMR2	0001_H	16	FFE6 $050E_H$	1(1H)	16
ATU4B	Multiplication Clock Counter B6	TCNTB6	0000 0000_H	32	FFE6 0510_H	1(1H)	32
ATU4B	Input Capture Register B6	ICRB6	0000 0000_H	32	FFE6 0514_H	1(1H)	32
ATU4B	Multiplication Setting Register B6	RARB6	40_H	8	FFE6 0518_H	1(1H)	8
ATU4B	Multiplication Clock Counter B6M	TCNTB6M	0000 0000_H	32	FFE6 $051C_H$	1(1H)	32
ATU4B	Output Comparison Register B6	OCRB6	FFFF $F000_H$	32	FFE6 0520_H	1(1H)	32
ATU4B	Output Comparison Register B7	OCRB7	FFFF $F000_H$	32	FFE6 0524_H	1(1H)	32
ATU4B	Correcting Event Counter B3	TCNTB3	0000 0000_H	32	FFE6 0580_H	1(1H)	32
ATU4B	Correcting Multiplication Clock Counter B4	TCNTB4	0000 0000_H	32	FFE6 0584_H	1(1H)	32
ATU4B	Multiplied and Corrected Clock Generating Counter B5	TCNTB5	0000 1000_H	32	FFE6 0588_H	1(1H)	32
ATU4B	Correcting Counter Clear Flag Register B	TCCLFRB	00_H	8	FFE6 $058C_H$	1(1H)	8
ATU4B	Correcting Counter Clear Flag Set Register B	TCCLFSRB	00_H	8	FFE6 $058D_H$	1(1H)	8
ATU4B	Correcting Counter Clear Flag Clear Register B	TCCLFCRB	00_H	8	FFE6 $058E_H$	1(1H)	8
ATU4B	Correcting Counter Clear Register B	TCCLRB	0000 0000_H	32	FFE6 0590_H	1(1H)	32
ATU4B	Output Comparison Register B8	OCRB8	FFFF $F000_H$	32	FFE6 0594_H	1(1H)	32
ATU4B	AGCKM2 Correcting Enable Setting Register	ACRTRGB	00_H	8	FFE6 $05A0_H$	1(1H)	8
ATU4B	AGCKM2 Correcting Clear Setting Register	ACRCLRB	00_H	8	FFE6 $05A1_H$	1(1H)	8
ATU4B	AGCKM2 Correcting Status Register	ACRSTRB	00_H	8	FFE6 $05A2_H$	1(1H)	8

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
ATU4B	AGCKM2 Correcting Clock Count Setting Register	ACRVALRB	0000_H	16	FFE6 $05A4_H$	1(1H)	16
ATU4C	Timer Start Register C	TSTRC	00_H	8	FFE6 0600_H	1(1H)	8
ATU4C	Noise Canceller Control Register C0	NCCRC0	00_H	8	FFE6 0604_H	1(1H)	8
ATU4C	Noise Canceller Control Register C1	NCCRC1	00_H	8	FFE6 0605_H	1(1H)	8
ATU4C	Noise Canceller Control Register C2	NCCRC2	00_H	8	FFE6 0606_H	1(1H)	8
ATU4C	Noise Canceller Control Register C3	NCCRC3	00_H	8	FFE6 0607_H	1(1H)	8
ATU4C	Noise Canceller Control Register C4	NCCRC4	00_H	8	FFE6 0608_H	1(1H)	8
ATU4C	Noise Canceller Control Register C5	NCCRC5	00_H	8	FFE6 0609_H	1(1H)	8
ATU4C	Noise Canceller Control Register C6	NCCRC6	00_H	8	FFE6 $060A_H$	1(1H)	8
ATU4C	Noise Canceller Control Register C7	NCCRC7	00_H	8	FFE6 $060B_H$	1(1H)	8
ATU4C	Noise Cancel Mode Channel Register 1C	NCMCR1C	00_H	8	FFE6 0610_H	1(1H)	8
ATU4C	Noise Cancel Mode Channel Register 2C	NCMCR2C	00_H	8	FFE6 0612_H	1(1H)	8
ATU4C	Timer Status Clear Register C0	TSCRC0	0000_H	16	FFE6 0622_H	1(1H)	8, 16
ATU4C	Timer Control Register C0	TCRC0	0000_H	16	FFE6 0624_H	1(1H)	8, 16
ATU4C	Timer Status Register C0	TSRC0	0000_H	16	FFE6 0626_H	1(1H)	8, 16
ATU4C	Timer I/O Control Register C0	TIORC0	0000_H	16	FFE6 0628_H	1(1H)	8, 16
ATU4C	Timer Interrupt Enable Register C0	TIERC0	$000F_H$	16	FFE6 0634_H	1(1H)	8, 16
ATU4C	Timer counter C0	TCNTC0	$0000\ 0000_H$	32	FFE6 0638_H	1(1H)	32
ATU4C	Counter Upper-Limit Setting Compare Register C0	CUCRC0	$FFFF\ FFFF_H$	32	FFE6 $063C_H$	1(1H)	32
ATU4C	Timer General Register C00	GRC00	$FFFF\ FFFF_H$	32	FFE6 0640_H	1(1H)	32
ATU4C	Timer General Register C01	GRC01	$FFFF\ FFFF_H$	32	FFE6 0644_H	1(1H)	32
ATU4C	Timer General Register C02	GRC02	$FFFF\ FFFF_H$	32	FFE6 0648_H	1(1H)	32
ATU4C	Timer General Register C03	GRC03	$FFFF\ FFFF_H$	32	FFE6 $064C_H$	1(1H)	32
ATU4C	Output Comparison Register C00	OCRC00	$FFFF\ FFFF_H$	32	FFE6 0650_H	1(1H)	32
ATU4C	Output Comparison Register C01	OCRC01	$FFFF\ FFFF_H$	32	FFE6 0654_H	1(1H)	32
ATU4C	Output Comparison Register C02	OCRC02	$FFFF\ FFFF_H$	32	FFE6 0658_H	1(1H)	32
ATU4C	Output Comparison Register C03	OCRC03	$FFFF\ FFFF_H$	32	FFE6 $065C_H$	1(1H)	32
ATU4C	Noise Cancel Counter C00	NCNTC00	0000_H	16	FFE6 0670_H	1(1H)	16
ATU4C	Noise cancel Register C00	NCRC00	0000_H	16	FFE6 0672_H	1(1H)	16
ATU4C	Noise Cancel Counter C01	NCNTC01	0000_H	16	FFE6 0674_H	1(1H)	16
ATU4C	Noise cancel Register C01	NCRC01	0000_H	16	FFE6 0676_H	1(1H)	16
ATU4C	Noise Cancel Counter C02	NCNTC02	0000_H	16	FFE6 0678_H	1(1H)	16
ATU4C	Noise cancel Register C02	NCRC02	0000_H	16	FFE6 $067A_H$	1(1H)	16
ATU4C	Noise Cancel Counter C03	NCNTC03	0000_H	16	FFE6 $067C_H$	1(1H)	16
ATU4C	Noise cancel Register C03	NCRC03	0000_H	16	FFE6 $067E_H$	1(1H)	16
ATU4C	Timer Status Clear Register C1	TSCRC1	0000_H	16	FFE6 $06A2_H$	1(1H)	8, 16
ATU4C	Timer Control Register C1	TCRC1	0000_H	16	FFE6 $06A4_H$	1(1H)	8, 16
ATU4C	Timer Status Register C1	TSRC1	0000_H	16	FFE6 $06A6_H$	1(1H)	8, 16
ATU4C	Timer I/O Control Register C1	TIORC1	0000_H	16	FFE6 $06A8_H$	1(1H)	8, 16
ATU4C	Timer Interrupt Enable Register C1	TIERC1	$000F_H$	16	FFE6 $06B4_H$	1(1H)	8, 16
ATU4C	Timer counter C1	TCNTC1	$0000\ 0000_H$	32	FFE6 $06B8_H$	1(1H)	32
ATU4C	Counter Upper-limit Setting Comparison Register C1	CUCRC1	$FFFF\ FFFF_H$	32	FFE6 $06BC_H$	1(1H)	32
ATU4C	Timer General Register C10	GRC10	$FFFF\ FFFF_H$	32	FFE6 $06C0_H$	1(1H)	32
ATU4C	Timer General Register C11	GRC11	$FFFF\ FFFF_H$	32	FFE6 $06C4_H$	1(1H)	32

(72/201)

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
ATU4C	Timer General Register C12	GRC12	FFFF FFFF$_H$	32	FFE6 06C8$_H$	1(1H)	32
ATU4C	Timer General Register C13	GRC13	FFFF FFFF$_H$	32	FFE6 06CC$_H$	1(1H)	32
ATU4C	Output Comparison Register C10	OCRC10	FFFF FFFF$_H$	32	FFE6 06D0$_H$	1(1H)	32
ATU4C	Output Comparison Register C11	OCRC11	FFFF FFFF$_H$	32	FFE6 06D4$_H$	1(1H)	32
ATU4C	Output Comparison Register C12	OCRC12	FFFF FFFF$_H$	32	FFE6 06D8$_H$	1(1H)	32
ATU4C	Output Comparison Register C13	OCRC13	FFFF FFFF$_H$	32	FFE6 06DC$_H$	1(1H)	32
ATU4C	Noise Cancel Counter C10	NCNTC10	0000$_H$	16	FFE6 06F0$_H$	1(1H)	16
ATU4C	Noise cancel Register C10	NCRC10	0000$_H$	16	FFE6 06F2$_H$	1(1H)	16
ATU4C	Noise Cancel Counter C11	NCNTC11	0000$_H$	16	FFE6 06F4$_H$	1(1H)	16
ATU4C	Noise cancel Register C11	NCRC11	0000$_H$	16	FFE6 06F6$_H$	1(1H)	16
ATU4C	Noise Cancel Counter C12	NCNTC12	0000$_H$	16	FFE6 06F8$_H$	1(1H)	16
ATU4C	Noise cancel Register C12	NCRC12	0000$_H$	16	FFE6 06FA$_H$	1(1H)	16
ATU4C	Noise Cancel Counter C13	NCNTC13	0000$_H$	16	FFE6 06FC$_H$	1(1H)	16
ATU4C	Noise cancel Register C13	NCRC13	0000$_H$	16	FFE6 06FE$_H$	1(1H)	16
ATU4C	Timer Status Clear Register C2	TSCRC2	0000$_H$	16	FFE6 0722$_H$	1(1H)	8, 16
ATU4C	Timer Control Register C2	TCRC2	0000$_H$	16	FFE6 0724$_H$	1(1H)	8, 16
ATU4C	Timer Status Register C2	TSRC2	0000$_H$	16	FFE6 0726$_H$	1(1H)	8, 16
ATU4C	Timer I/O Control Register C2	TIORC2	0000$_H$	16	FFE6 0728$_H$	1(1H)	8, 16
ATU4C	Timer Interrupt Enable Register C2	TIERC2	000F$_H$	16	FFE6 0734$_H$	1(1H)	8, 16
ATU4C	Timer counter C2	TCNTC2	0000 0000$_H$	32	FFE6 0738$_H$	1(1H)	32
ATU4C	Counter Upper-limit Setting Comparison Register C2	CUCRC2	FFFF FFFF$_H$	32	FFE6 073C$_H$	1(1H)	32
ATU4C	Timer General Register C20	GRC20	FFFF FFFF$_H$	32	FFE6 0740$_H$	1(1H)	32
ATU4C	Timer General Register C21	GRC21	FFFF FFFF$_H$	32	FFE6 0744$_H$	1(1H)	32
ATU4C	Timer General Register C22	GRC22	FFFF FFFF$_H$	32	FFE6 0748$_H$	1(1H)	32
ATU4C	Timer General Register C23	GRC23	FFFF FFFF$_H$	32	FFE6 074C$_H$	1(1H)	32
ATU4C	Output Comparison Register C20	OCRC20	FFFF FFFF$_H$	32	FFE6 0750$_H$	1(1H)	32
ATU4C	Output Comparison Register C21	OCRC21	FFFF FFFF$_H$	32	FFE6 0754$_H$	1(1H)	32
ATU4C	Output Comparison Register C22	OCRC22	FFFF FFFF$_H$	32	FFE6 0758$_H$	1(1H)	32
ATU4C	Output Comparison Register C23	OCRC23	FFFF FFFF$_H$	32	FFE6 075C$_H$	1(1H)	32
ATU4C	Noise Cancel Counter C20	NCNTC20	0000$_H$	16	FFE6 0770$_H$	1(1H)	16
ATU4C	Noise cancel Register C20	NCRC20	0000$_H$	16	FFE6 0772$_H$	1(1H)	16
ATU4C	Noise Cancel Counter C21	NCNTC21	0000$_H$	16	FFE6 0774$_H$	1(1H)	16
ATU4C	Noise cancel Register C21	NCRC21	0000$_H$	16	FFE6 0776$_H$	1(1H)	16
ATU4C	Noise Cancel Counter C22	NCNTC22	0000$_H$	16	FFE6 0778$_H$	1(1H)	16
ATU4C	Noise cancel Register C22	NCRC22	0000$_H$	16	FFE6 077A$_H$	1(1H)	16
ATU4C	Noise Cancel Counter C23	NCNTC23	0000$_H$	16	FFE6 077C$_H$	1(1H)	16
ATU4C	Noise cancel Register C23	NCRC23	0000$_H$	16	FFE6 077E$_H$	1(1H)	16
ATU4C	Timer Status Clear Register C3	TSCRC3	0000$_H$	16	FFE6 07A2$_H$	1(1H)	8, 16
ATU4C	Timer Control Register C3	TCRC3	0000$_H$	16	FFE6 07A4$_H$	1(1H)	8, 16
ATU4C	Timer Status Register C3	TSRC3	0000$_H$	16	FFE6 07A6$_H$	1(1H)	8, 16
ATU4C	Timer I/O Control Register C3	TIORC3	0000$_H$	16	FFE6 07A8$_H$	1(1H)	8, 16
ATU4C	Timer Interrupt Enable Register C3	TIERC3	000F$_H$	16	FFE6 07B4$_H$	1(1H)	8, 16
ATU4C	Timer Counter C3	TCNTC3	0000 0000$_H$	32	FFE6 07B8$_H$	1(1H)	32
ATU4C	Counter Upper-limit Setting Comparison Register C3	CUCRC3	FFFF FFFF$_H$	32	FFE6 07BC$_H$	1(1H)	32
ATU4C	Timer General Register C30	GRC30	FFFF FFFF$_H$	32	FFE6 07C0$_H$	1(1H)	32
ATU4C	Timer General Register C31	GRC31	FFFF FFFF$_H$	32	FFE6 07C4$_H$	1(1H)	32

(73/201)

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
ATU4C	Timer General Register C32	GRC32	FFFF FFFF$_H$	32	FFE6 07C8$_H$	1(1H)	32
ATU4C	Timer General Register C33	GRC33	FFFF FFFF$_H$	32	FFE6 07CC$_H$	1(1H)	32
ATU4C	Output Comparison Register C30	OCRC30	FFFF FFFF$_H$	32	FFE6 07D0$_H$	1(1H)	32
ATU4C	Output Comparison Register C31	OCRC31	FFFF FFFF$_H$	32	FFE6 07D4$_H$	1(1H)	32
ATU4C	Output Comparison Register C32	OCRC32	FFFF FFFF$_H$	32	FFE6 07D8$_H$	1(1H)	32
ATU4C	Output Comparison Register C33	OCRC33	FFFF FFFF$_H$	32	FFE6 07DC$_H$	1(1H)	32
ATU4C	Noise Cancel Counter C30	NCNTC30	0000$_H$	16	FFE6 07F0$_H$	1(1H)	16
ATU4C	Noise cancel Register C30	NCRC30	0000$_H$	16	FFE6 07F2$_H$	1(1H)	16
ATU4C	Noise Cancel Counter C31	NCNTC31	0000$_H$	16	FFE6 07F4$_H$	1(1H)	16
ATU4C	Noise cancel Register C31	NCRC31	0000$_H$	16	FFE6 07F6$_H$	1(1H)	16
ATU4C	Noise Cancel Counter C32	NCNTC32	0000$_H$	16	FFE6 07F8$_H$	1(1H)	16
ATU4C	Noise cancel Register C32	NCRC32	0000$_H$	16	FFE6 07FA$_H$	1(1H)	16
ATU4C	Noise Cancel Counter C33	NCNTC33	0000$_H$	16	FFE6 07FC$_H$	1(1H)	16
ATU4C	Noise cancel Register C33	NCRC33	0000$_H$	16	FFE6 07FE$_H$	1(1H)	16
ATU4C	Timer Status Clear Register C4	TSCRC4	0000$_H$	16	FFE6 0822$_H$	1(1H)	8, 16
ATU4C	Timer Control Register C4	TCRC4	0000$_H$	16	FFE6 0824$_H$	1(1H)	8, 16
ATU4C	Timer Status Register C4	TSRC4	0000$_H$	16	FFE6 0826$_H$	1(1H)	8, 16
ATU4C	Timer I/O Control Register C4	TIORC4	0000$_H$	16	FFE6 0828$_H$	1(1H)	8, 16
ATU4C	Timer Interrupt Enable Register C4	TIERC4	000F$_H$	16	FFE6 0834$_H$	1(1H)	8, 16
ATU4C	Timer counter C4	TCNTC4	0000 0000$_H$	32	FFE6 0838$_H$	1(1H)	32
ATU4C	Counter Upper-limit Setting Comparison Register C4	CUCRC4	FFFF FFFF$_H$	32	FFE6 083C$_H$	1(1H)	32
ATU4C	Timer General Register C40	GRC40	FFFF FFFF$_H$	32	FFE6 0840$_H$	1(1H)	32
ATU4C	Timer General Register C41	GRC41	FFFF FFFF$_H$	32	FFE6 0844$_H$	1(1H)	32
ATU4C	Timer General Register C42	GRC42	FFFF FFFF$_H$	32	FFE6 0848$_H$	1(1H)	32
ATU4C	Timer General Register C43	GRC43	FFFF FFFF$_H$	32	FFE6 084C$_H$	1(1H)	32
ATU4C	Output Comparison Register C40	OCRC40	FFFF FFFF$_H$	32	FFE6 0850$_H$	1(1H)	32
ATU4C	Output Comparison Register C41	OCRC41	FFFF FFFF$_H$	32	FFE6 0854$_H$	1(1H)	32
ATU4C	Output Comparison Register C42	OCRC42	FFFF FFFF$_H$	32	FFE6 0858$_H$	1(1H)	32
ATU4C	Output Comparison Register C43	OCRC43	FFFF FFFF$_H$	32	FFE6 085C$_H$	1(1H)	32
ATU4C	Noise Cancel Counter C40	NCNTC40	0000$_H$	16	FFE6 0870$_H$	1(1H)	16
ATU4C	Noise cancel Register C40	NCRC40	0000$_H$	16	FFE6 0872$_H$	1(1H)	16
ATU4C	Noise Cancel Counter C41	NCNTC41	0000$_H$	16	FFE6 0874$_H$	1(1H)	16
ATU4C	Noise cancel Register C41	NCRC41	0000$_H$	16	FFE6 0876$_H$	1(1H)	16
ATU4C	Noise Cancel Counter C42	NCNTC42	0000$_H$	16	FFE6 0878$_H$	1(1H)	16
ATU4C	Noise cancel Register C42	NCRC42	0000$_H$	16	FFE6 087A$_H$	1(1H)	16
ATU4C	Noise Cancel Counter C43	NCNTC43	0000$_H$	16	FFE6 087C$_H$	1(1H)	16
ATU4C	Noise cancel Register C43	NCRC43	0000$_H$	16	FFE6 087E$_H$	1(1H)	16
ATU4C	Timer Status Clear Register C5	TSCRC5	0000$_H$	16	FFE6 08A2$_H$	1(1H)	8, 16
ATU4C	Timer Control Register C5	TCRC5	0000$_H$	16	FFE6 08A4$_H$	1(1H)	8, 16
ATU4C	Timer Status Register C5	TSRC5	0000$_H$	16	FFE6 08A6$_H$	1(1H)	8, 16
ATU4C	Timer I/O Control Register C5	TIORC5	0000$_H$	16	FFE6 08A8$_H$	1(1H)	8, 16
ATU4C	Timer Interrupt Enable Register C5	TIERC5	000F$_H$	16	FFE6 08B4$_H$	1(1H)	8, 16
ATU4C	Timer counter C5	TCNTC5	0000 0000$_H$	32	FFE6 08B8$_H$	1(1H)	32
ATU4C	Counter Upper-limit Setting Comparison Register C5	CUCRC5	FFFF FFFF$_H$	32	FFE6 08BC$_H$	1(1H)	32
ATU4C	Timer General Register C50	GRC50	FFFF FFFF$_H$	32	FFE6 08C0$_H$	1(1H)	32
ATU4C	Timer General Register C51	GRC51	FFFF FFFF$_H$	32	FFE6 08C4$_H$	1(1H)	32

(74/201)

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
ATU4C	Timer General Register C52	GRC52	FFFF FFFF$_H$	32	FFE6 08C8$_H$	1(1H)	32
ATU4C	Timer General Register C53	GRC53	FFFF FFFF$_H$	32	FFE6 08CC$_H$	1(1H)	32
ATU4C	Output Comparison Register C50	OCRC50	FFFF FFFF$_H$	32	FFE6 08D0$_H$	1(1H)	32
ATU4C	Output Comparison Register C51	OCRC51	FFFF FFFF$_H$	32	FFE6 08D4$_H$	1(1H)	32
ATU4C	Output Comparison Register C52	OCRC52	FFFF FFFF$_H$	32	FFE6 08D8$_H$	1(1H)	32
ATU4C	Output Comparison Register C53	OCRC53	FFFF FFFF$_H$	32	FFE6 08DC$_H$	1(1H)	32
ATU4C	Noise Cancel Counter C50	NCNTC50	0000$_H$	16	FFE6 08F0$_H$	1(1H)	16
ATU4C	Noise cancel Register C50	NCRC50	0000$_H$	16	FFE6 08F2$_H$	1(1H)	16
ATU4C	Noise Cancel Counter C51	NCNTC51	0000$_H$	16	FFE6 08F4$_H$	1(1H)	16
ATU4C	Noise cancel Register C51	NCRC51	0000$_H$	16	FFE6 08F6$_H$	1(1H)	16
ATU4C	Noise Cancel Counter C52	NCNTC52	0000$_H$	16	FFE6 08F8$_H$	1(1H)	16
ATU4C	Noise cancel Register C52	NCRC52	0000$_H$	16	FFE6 08FA$_H$	1(1H)	16
ATU4C	Noise Cancel Counter C53	NCNTC53	0000$_H$	16	FFE6 08FC$_H$	1(1H)	16
ATU4C	Noise cancel Register C53	NCRC53	0000$_H$	16	FFE6 08FE$_H$	1(1H)	16
ATU4C	Timer Status Clear Register C6	TSCRC6	0000$_H$	16	FFE6 0922$_H$	1(1H)	8, 16
ATU4C	Timer Control Register C6	TCRC6	0000$_H$	16	FFE6 0924$_H$	1(1H)	8, 16
ATU4C	Timer Status Register C6	TSRC6	0000$_H$	16	FFE6 0926$_H$	1(1H)	8, 16
ATU4C	Timer I/O Control Register C6	TIORC6	0000$_H$	16	FFE6 0928$_H$	1(1H)	8, 16
ATU4C	Timer Interrupt Enable Register C6	TIERC6	000F$_H$	16	FFE6 0934$_H$	1(1H)	8, 16
ATU4C	Timer counter C6	TCNTC6	0000 0000$_H$	32	FFE6 0938$_H$	1(1H)	32
ATU4C	Counter Upper-limit Setting Comparison Register C6	CUCRC6	FFFF FFFF$_H$	32	FFE6 093C$_H$	1(1H)	32
ATU4C	Timer General Register C60	GRC60	FFFF FFFF$_H$	32	FFE6 0940$_H$	1(1H)	32
ATU4C	Timer General Register C61	GRC61	FFFF FFFF$_H$	32	FFE6 0944$_H$	1(1H)	32
ATU4C	Timer General Register C62	GRC62	FFFF FFFF$_H$	32	FFE6 0948$_H$	1(1H)	32
ATU4C	Timer General Register C63	GRC63	FFFF FFFF$_H$	32	FFE6 094C$_H$	1(1H)	32
ATU4C	Output Comparison Register C60	OCRC60	FFFF FFFF$_H$	32	FFE6 0950$_H$	1(1H)	32
ATU4C	Output Comparison Register C61	OCRC61	FFFF FFFF$_H$	32	FFE6 0954$_H$	1(1H)	32
ATU4C	Output Comparison Register C62	OCRC62	FFFF FFFF$_H$	32	FFE6 0958$_H$	1(1H)	32
ATU4C	Output Comparison Register C63	OCRC63	FFFF FFFF$_H$	32	FFE6 095C$_H$	1(1H)	32
ATU4C	Noise Cancel Counter C60	NCNTC60	0000$_H$	16	FFE6 0970$_H$	1(1H)	16
ATU4C	Noise cancel Register C60	NCRC60	0000$_H$	16	FFE6 0972$_H$	1(1H)	16
ATU4C	Noise Cancel Counter C61	NCNTC61	0000$_H$	16	FFE6 0974$_H$	1(1H)	16
ATU4C	Noise cancel Register C61	NCRC61	0000$_H$	16	FFE6 0976$_H$	1(1H)	16
ATU4C	Noise Cancel Counter C62	NCNTC62	0000$_H$	16	FFE6 0978$_H$	1(1H)	16
ATU4C	Noise cancel Register C62	NCRC62	0000$_H$	16	FFE6 097A$_H$	1(1H)	16
ATU4C	Noise Cancel Counter C63	NCNTC63	0000$_H$	16	FFE6 097C$_H$	1(1H)	16
ATU4C	Noise cancel Register C63	NCRC63	0000$_H$	16	FFE6 097E$_H$	1(1H)	16
ATU4C	Timer Status Clear Register C7	TSCRC7	0000$_H$	16	FFE6 09A2$_H$	1(1H)	8, 16
ATU4C	Timer Control Register C7	TCRC7	0000$_H$	16	FFE6 09A4$_H$	1(1H)	8, 16
ATU4C	Timer Status Register C7	TSRC7	0000$_H$	16	FFE6 09A6$_H$	1(1H)	8, 16
ATU4C	Timer I/O Control Register C7	TIORC7	0000$_H$	16	FFE6 09A8$_H$	1(1H)	8, 16
ATU4C	Timer Interrupt Enable Register C7	TIERC7	000F$_H$	16	FFE6 09B4$_H$	1(1H)	8, 16
ATU4C	Timer counter C7	TCNTC7	0000 0000$_H$	32	FFE6 09B8$_H$	1(1H)	32
ATU4C	Counter Upper-limit Setting Comparison Register C7	CUCRC7	FFFF FFFF$_H$	32	FFE6 09BC$_H$	1(1H)	32
ATU4C	Timer General Register C70	GRC70	FFFF FFFF$_H$	32	FFE6 09C0$_H$	1(1H)	32
ATU4C	Timer General Register C71	GRC71	FFFF FFFF$_H$	32	FFE6 09C4$_H$	1(1H)	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
ATU4C	Timer General Register C72	GRC72	FFFF FFFF$_H$	32	FFE6 09C8$_H$	1(1H)	32
ATU4C	Timer General Register C73	GRC73	FFFF FFFF$_H$	32	FFE6 09CC$_H$	1(1H)	32
ATU4C	Output Comparison Register C70	OCRC70	FFFF FFFF$_H$	32	FFE6 09D0$_H$	1(1H)	32
ATU4C	Output Comparison Register C71	OCRC71	FFFF FFFF$_H$	32	FFE6 09D4$_H$	1(1H)	32
ATU4C	Output Comparison Register C72	OCRC72	FFFF FFFF$_H$	32	FFE6 09D8$_H$	1(1H)	32
ATU4C	Output Comparison Register C73	OCRC73	FFFF FFFF$_H$	32	FFE6 09DC$_H$	1(1H)	32
ATU4C	Noise Cancel Counter C70	NCNTC70	0000$_H$	16	FFE6 09F0$_H$	1(1H)	16
ATU4C	Noise cancel Register C70	NCRC70	0000$_H$	16	FFE6 09F2$_H$	1(1H)	16
ATU4C	Noise Cancel Counter C71	NCNTC71	0000$_H$	16	FFE6 09F4$_H$	1(1H)	16
ATU4C	Noise cancel Register C71	NCRC71	0000$_H$	16	FFE6 09F6$_H$	1(1H)	16
ATU4C	Noise Cancel Counter C72	NCNTC72	0000$_H$	16	FFE6 09F8$_H$	1(1H)	16
ATU4C	Noise cancel Register C72	NCRC72	0000$_H$	16	FFE6 09FA$_H$	1(1H)	16
ATU4C	Noise Cancel Counter C73	NCNTC73	0000$_H$	16	FFE6 09FC$_H$	1(1H)	16
ATU4C	Noise cancel Register C73	NCRC73	0000$_H$	16	FFE6 09FE$_H$	1(1H)	16
ATU4D	Timer Start Register D	TSTRD	0000$_H$	16	FFE6 1000$_H$	1(1H)	8, 16
ATU4D	Timer Control Register D0	TCRD0	0000$_H$	16	FFE6 1100$_H$	1(1H)	8, 16
ATU4D	Down Counter Control Register D0	DCRD0	0000$_H$	16	FFE6 1102$_H$	1(1H)	8, 16
ATU4D	Timer I/O Control Register 1D0	TIOR1D0	0000$_H$	16	FFE6 1104$_H$	1(1H)	8, 16
ATU4D	Timer I/O Control Register 2D0	TIOR2D0	0000$_H$	16	FFE6 1106$_H$	1(1H)	8, 16
ATU4D	Down Count Start Register D0	DSTRD0	00$_H$	8	FFE6 1108$_H$	1(1H)	8
ATU4D	Down Count Status Clear Register D0	DSCRD0	00$_H$	8	FFE6 1109$_H$	1(1H)	8
ATU4D	Down Count Status Register 1D0	DSR1D0	00$_H$	8	FFE6 110A$_H$	1(1H)	8
ATU4D	Down Count Status Register 2D0	DSR2D0	00$_H$	8	FFE6 110B$_H$	1(1H)	8
ATU4D	Timer Status Register D0	TSRD0	0000$_H$	16	FFE6 110C$_H$	1(1H)	8, 16
ATU4D	Timer Status Clear Register D0	TSCRD0	0000$_H$	16	FFE6 110E$_H$	1(1H)	8, 16
ATU4D	Timer Output Control Register D0	TOCRD0	00$_H$	8	FFE6 1110$_H$	1(1H)	8
ATU4D	Output Selection Register D0	OSELRD0	00$_H$	8	FFE6 1111$_H$	1(1H)	8
ATU4D	Output Value Register D0	ODRD0	00$_H$	8	FFE6 1112$_H$	1(1H)	8
ATU4D	Timer Input Capture Trigger Selection Register D0	TICTSELD0	00$_H$	8	FFE6 1114$_H$	1(1H)	8
ATU4D	Timer Offset Base Register D0	OSBRD0	0000 0000$_H$	32	FFE6 1118$_H$	1(1H)	32
ATU4D	Timer counter 1D0	TCNT1D0	0000 0000$_H$	32	FFE6 1120$_H$	1(1H)	32
ATU4D	Timer counter 2D0	TCNT2D0	0000 0000$_H$	32	FFE6 1124$_H$	1(1H)	32
ATU4D	Counter Upper-limit Setting Comparison Register 1 D0	CUCR1D0	FFFF FFFF$_H$	32	FFE6 1128$_H$	1(1H)	32
ATU4D	Counter Upper-limit Setting Comparison Register 2 D0	CUCR2D0	FFFF FFFF$_H$	32	FFE6 112C$_H$	1(1H)	32
ATU4D	Range Comparison Value Setting Register 1 D0	RCR1D0	00$_H$	8	FFE6 1130$_H$	1(1H)	8
ATU4D	Range Comparison Value Setting Register 2 D0	RCR2D0	00$_H$	8	FFE6 1134$_H$	1(1H)	8
ATU4D	Output Comparison Register 1 D00	OCR1D00	FFFF FFFF$_H$	32	FFE6 1140$_H$	1(1H)	32
ATU4D	Output Comparison Register 2 D00	OCR2D00	FFFF FFFF$_H$	32	FFE6 1148$_H$	1(1H)	32
ATU4D	Input Capture Register 1 D00	ICR1D00	0000 0000$_H$	32	FFE6 1150$_H$	1(1H)	32
ATU4D	Input Capture Register 2 D00	ICR2D00	0000 0000$_H$	32	FFE6 1154$_H$	1(1H)	32
ATU4D	Timer Down Counter D00	DCNTD00	0000 0000$_H$	32	FFE6 1158$_H$	1(1H)	32
ATU4D	Output Comparison Register 1 D01	OCR1D01	FFFF FFFF$_H$	32	FFE6 1160$_H$	1(1H)	32
ATU4D	Output Comparison Register 2 D01	OCR2D01	FFFF FFFF$_H$	32	FFE6 1168$_H$	1(1H)	32
ATU4D	Input Capture Register 1 D01	ICR1D01	0000 0000$_H$	32	FFE6 1170$_H$	1(1H)	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
ATU4D	Input Capture Register 2 D01	ICR2D01	0000 0000$_H$	32	FFE6 1174$_H$	1(1H)	32
ATU4D	Timer Down Counter D01	DCNTD01	0000 0000$_H$	32	FFE6 1178$_H$	1(1H)	32
ATU4D	Output Comparison Register 1 D02	OCR1D02	FFFF FFFF$_H$	32	FFE6 1180$_H$	1(1H)	32
ATU4D	Output Comparison Register 2 D02	OCR2D02	FFFF FFFF$_H$	32	FFE6 1188$_H$	1(1H)	32
ATU4D	Input Capture Register 1 D02	ICR1D02	0000 0000$_H$	32	FFE6 1190$_H$	1(1H)	32
ATU4D	Input Capture Register 2 D02	ICR2D02	0000 0000$_H$	32	FFE6 1194$_H$	1(1H)	32
ATU4D	Timer Down Counter D02	DCNTD02	0000 0000$_H$	32	FFE6 1198$_H$	1(1H)	32
ATU4D	Output Comparison Register 1 D03	OCR1D03	FFFF FFFF$_H$	32	FFE6 11A0$_H$	1(1H)	32
ATU4D	Output Comparison Register 2 D03	OCR2D03	FFFF FFFF$_H$	32	FFE6 11A8$_H$	1(1H)	32
ATU4D	Input Capture Register 1 D03	ICR1D03	0000 0000$_H$	32	FFE6 11B0$_H$	1(1H)	32
ATU4D	Input Capture Register 2 D03	ICR2D03	0000 0000$_H$	32	FFE6 11B4$_H$	1(1H)	32
ATU4D	Timer Down Counter D03	DCNTD03	0000 0000$_H$	32	FFE6 11B8$_H$	1(1H)	32
ATU4D	Timer Control Register D1	TCRD1	0000$_H$	16	FFE6 1200$_H$	1(1H)	8, 16
ATU4D	Down Counter Control Register D1	DCRD1	0000$_H$	16	FFE6 1202$_H$	1(1H)	8, 16
ATU4D	Timer I/O Control Register 1 D1	TIOR1D1	0000$_H$	16	FFE6 1204$_H$	1(1H)	8, 16
ATU4D	Timer I/O Control Register 2 D1	TIOR2D1	0000$_H$	16	FFE6 1206$_H$	1(1H)	8, 16
ATU4D	Down Count Start Register D1	DSTRD1	00$_H$	8	FFE6 1208$_H$	1(1H)	8
ATU4D	Down Count Status Clear Register D1	DSCRD1	00$_H$	8	FFE6 1209$_H$	1(1H)	8
ATU4D	Down Count Status Register 1 D1	DSR1D1	00$_H$	8	FFE6 120A$_H$	1(1H)	8
ATU4D	Down Count Status Register 2 D1	DSR2D1	00$_H$	8	FFE6 120B$_H$	1(1H)	8
ATU4D	Timer Status Register D1	TSRD1	0000$_H$	16	FFE6 120C$_H$	1(1H)	8, 16
ATU4D	Timer Status Clear Register D1	TSCRD1	0000$_H$	16	FFE6 120E$_H$	1(1H)	8, 16
ATU4D	Timer Output Control Register D1	TOCRD1	00$_H$	8	FFE6 1210$_H$	1(1H)	8
ATU4D	Output Selection Register D1	OSELRD1	00$_H$	8	FFE6 1211$_H$	1(1H)	8
ATU4D	Output Value Register D1	ODRD1	00$_H$	8	FFE6 1212$_H$	1(1H)	8
ATU4D	Timer Input Capture Trigger Selection Register D1	TICTSELD1	00$_H$	8	FFE6 1214$_H$	1(1H)	8
ATU4D	Timer Offset Base Register D1	OSBRD1	0000 0000$_H$	32	FFE6 1218$_H$	1(1H)	32
ATU4D	Timer counter 1 D1	TCNT1D1	0000 0000$_H$	32	FFE6 1220$_H$	1(1H)	32
ATU4D	Timer counter 2 D1	TCNT2D1	0000 0000$_H$	32	FFE6 1224$_H$	1(1H)	32
ATU4D	Counter Upper-limit Setting Comparison Register 1 D1	CUCR1D1	FFFF FFFF$_H$	32	FFE6 1228$_H$	1(1H)	32
ATU4D	Counter Upper-limit Setting Comparison Register 2 D1	CUCR2D1	FFFF FFFF$_H$	32	FFE6 122C$_H$	1(1H)	32
ATU4D	Range Comparison Value Setting Register 1 D1	RCR1D1	00$_H$	8	FFE6 1230$_H$	1(1H)	8
ATU4D	Range Comparison Value Setting Register 2 D1	RCR2D1	00$_H$	8	FFE6 1234$_H$	1(1H)	8
ATU4D	Output Comparison Register 1 D10	OCR1D10	FFFF FFFF$_H$	32	FFE6 1240$_H$	1(1H)	32
ATU4D	Output Comparison Register 2 D10	OCR2D10	FFFF FFFF$_H$	32	FFE6 1248$_H$	1(1H)	32
ATU4D	Input Capture Register 1 D10	ICR1D10	0000 0000$_H$	32	FFE6 1250$_H$	1(1H)	32
ATU4D	Input Capture Register 2 D10	ICR2D10	0000 0000$_H$	32	FFE6 1254$_H$	1(1H)	32
ATU4D	Timer Down Counter D10	DCNTD10	0000 0000$_H$	32	FFE6 1258$_H$	1(1H)	32
ATU4D	Output Comparison Register 1 D11	OCR1D11	FFFF FFFF$_H$	32	FFE6 1260$_H$	1(1H)	32
ATU4D	Output Comparison Register 2 D11	OCR2D11	FFFF FFFF$_H$	32	FFE6 1268$_H$	1(1H)	32
ATU4D	Input Capture Register 1 D11	ICR1D11	0000 0000$_H$	32	FFE6 1270$_H$	1(1H)	32
ATU4D	Input Capture Register 2 D11	ICR2D11	0000 0000$_H$	32	FFE6 1274$_H$	1(1H)	32
ATU4D	Timer Down Counter D11	DCNTD11	0000 0000$_H$	32	FFE6 1278$_H$	1(1H)	32
ATU4D	Output Comparison Register 1 D12	OCR1D12	FFFF FFFF$_H$	32	FFE6 1280$_H$	1(1H)	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
ATU4D	Output Comparison Register 2 D12	OCR2D12	FFFF FFFF$_H$	32	FFE6 1288$_H$	1(1H)	32
ATU4D	Input Capture Register 1 D12	ICR1D12	0000 0000$_H$	32	FFE6 1290$_H$	1(1H)	32
ATU4D	Input Capture Register 2 D12	ICR2D12	0000 0000$_H$	32	FFE6 1294$_H$	1(1H)	32
ATU4D	Timer Down Counter D12	DCNTD12	0000 0000$_H$	32	FFE6 1298$_H$	1(1H)	32
ATU4D	Output Comparison Register 1 D13	OCR1D13	FFFF FFFF$_H$	32	FFE6 12A0$_H$	1(1H)	32
ATU4D	Output Comparison Register 2 D13	OCR2D13	FFFF FFFF$_H$	32	FFE6 12A8$_H$	1(1H)	32
ATU4D	Input Capture Register 1 D13	ICR1D13	0000 0000$_H$	32	FFE6 12B0$_H$	1(1H)	32
ATU4D	Input Capture Register 2 D13	ICR2D13	0000 0000$_H$	32	FFE6 12B4$_H$	1(1H)	32
ATU4D	Timer Down Counter D13	DCNTD13	0000 0000$_H$	32	FFE6 12B8$_H$	1(1H)	32
ATU4D	Timer Control Register D2	TCRD2	0000$_H$	16	FFE6 1300$_H$	1(1H)	8, 16
ATU4D	Down Counter Control Register D2	DCRD2	0000$_H$	16	FFE6 1302$_H$	1(1H)	8, 16
ATU4D	Timer I/O Control Register 1 D2	TIOR1D2	0000$_H$	16	FFE6 1304$_H$	1(1H)	8, 16
ATU4D	Timer I/O Control Register 2 D2	TIOR2D2	0000$_H$	16	FFE6 1306$_H$	1(1H)	8, 16
ATU4D	Down Count Start Register D2	DSTRD2	00$_H$	8	FFE6 1308$_H$	1(1H)	8
ATU4D	Down Count Status Clear Register D2	DSCRD2	00$_H$	8	FFE6 1309$_H$	1(1H)	8
ATU4D	Down Count Status Register 1 D2	DSR1D2	00$_H$	8	FFE6 130A$_H$	1(1H)	8
ATU4D	Down Count Status Register 2 D2	DSR2D2	00$_H$	8	FFE6 130B$_H$	1(1H)	8
ATU4D	Timer Status Register D2	TSRD2	0000$_H$	16	FFE6 130C$_H$	1(1H)	8, 16
ATU4D	Timer Status Clear Register D2	TSCRD2	0000$_H$	16	FFE6 130E$_H$	1(1H)	8, 16
ATU4D	Timer Output Control Register D2	TOCRD2	00$_H$	8	FFE6 1310$_H$	1(1H)	8
ATU4D	Output Selection Register D2	OSELRD2	00$_H$	8	FFE6 1311$_H$	1(1H)	8
ATU4D	Output Value Register D2	ODRD2	00$_H$	8	FFE6 1312$_H$	1(1H)	8
ATU4D	Timer Input Capture Trigger Selection Register D2	TICTSELD2	00$_H$	8	FFE6 1314$_H$	1(1H)	8
ATU4D	Timer Offset Base Register D2	OSBRD2	0000 0000$_H$	32	FFE6 1318$_H$	1(1H)	32
ATU4D	Timer counter 1 D2	TCNT1D2	0000 0000$_H$	32	FFE6 1320$_H$	1(1H)	32
ATU4D	Timer counter 2 D2	TCNT2D2	0000 0000$_H$	32	FFE6 1324$_H$	1(1H)	32
ATU4D	Counter Upper-limit Setting Comparison Register 1 D2	CUCR1D2	FFFF FFFF$_H$	32	FFE6 1328$_H$	1(1H)	32
ATU4D	Counter Upper-limit Setting Comparison Register 2 D2	CUCR2D2	FFFF FFFF$_H$	32	FFE6 132C$_H$	1(1H)	32
ATU4D	Range Comparison Value Setting Register 1 D2	RCR1D2	00$_H$	8	FFE6 1330$_H$	1(1H)	8
ATU4D	Range Comparison Value Setting Register 2 D2	RCR2D2	00$_H$	8	FFE6 1334$_H$	1(1H)	8
ATU4D	Output Comparison Register 1 D20	OCR1D20	FFFF FFFF$_H$	32	FFE6 1340$_H$	1(1H)	32
ATU4D	Output Comparison Register 2 D20	OCR2D20	FFFF FFFF$_H$	32	FFE6 1348$_H$	1(1H)	32
ATU4D	Input Capture Register 1 D20	ICR1D20	0000 0000$_H$	32	FFE6 1350$_H$	1(1H)	32
ATU4D	Input Capture Register 2 D20	ICR2D20	0000 0000$_H$	32	FFE6 1354$_H$	1(1H)	32
ATU4D	Timer Down Counter D20	DCNTD20	0000 0000$_H$	32	FFE6 1358$_H$	1(1H)	32
ATU4D	Output Comparison Register 1 D21	OCR1D21	FFFF FFFF$_H$	32	FFE6 1360$_H$	1(1H)	32
ATU4D	Output Comparison Register 2 D21	OCR2D21	FFFF FFFF$_H$	32	FFE6 1368$_H$	1(1H)	32
ATU4D	Input Capture Register 1 D21	ICR1D21	0000 0000$_H$	32	FFE6 1370$_H$	1(1H)	32
ATU4D	Input Capture Register 2 D21	ICR2D21	0000 0000$_H$	32	FFE6 1374$_H$	1(1H)	32
ATU4D	Timer Down Counter D21	DCNTD21	0000 0000$_H$	32	FFE6 1378$_H$	1(1H)	32
ATU4D	Output Comparison Register 1 D22	OCR1D22	FFFF FFFF$_H$	32	FFE6 1380$_H$	1(1H)	32
ATU4D	Output Comparison Register 2 D22	OCR2D22	FFFF FFFF$_H$	32	FFE6 1388$_H$	1(1H)	32
ATU4D	Input Capture Register 1 D22	ICR1D22	0000 0000$_H$	32	FFE6 1390$_H$	1(1H)	32
ATU4D	Input Capture Register 2 D22	ICR2D22	0000 0000$_H$	32	FFE6 1394$_H$	1(1H)	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
ATU4D	Timer Down Counter D22	DCNTD22	0000 0000$_H$	32	FFE6 1398$_H$	1(1H)	32
ATU4D	Output Comparison Register 1 D23	OCR1D23	FFFF FFFF$_H$	32	FFE6 13A0$_H$	1(1H)	32
ATU4D	Output Comparison Register 2 D23	OCR2D23	FFFF FFFF$_H$	32	FFE6 13A8$_H$	1(1H)	32
ATU4D	Input Capture Register 1 D23	ICR1D23	0000 0000$_H$	32	FFE6 13B0$_H$	1(1H)	32
ATU4D	Input Capture Register 2 D23	ICR2D23	0000 0000$_H$	32	FFE6 13B4$_H$	1(1H)	32
ATU4D	Timer Down Counter D23	DCNTD23	0000 0000$_H$	32	FFE6 13B8$_H$	1(1H)	32
ATU4D	Timer Control Register D3	TCRD3	0000$_H$	16	FFE6 1400$_H$	1(1H)	8, 16
ATU4D	Down Counter Control Register D3	DCRD3	0000$_H$	16	FFE6 1402$_H$	1(1H)	8, 16
ATU4D	Timer I/O Control Register 1 D3	TIOR1D3	0000$_H$	16	FFE6 1404$_H$	1(1H)	8, 16
ATU4D	Timer I/O Control Register 2 D3	TIOR2D3	0000$_H$	16	FFE6 1406$_H$	1(1H)	8, 16
ATU4D	Down Count Start Register D3	DSTRD3	00$_H$	8	FFE6 1408$_H$	1(1H)	8
ATU4D	Down Count Status Clear Register D3	DSCRD3	00$_H$	8	FFE6 1409$_H$	1(1H)	8
ATU4D	Down Count Status Register 1 D3	DSR1D3	00$_H$	8	FFE6 140A$_H$	1(1H)	8
ATU4D	Down Count Status Register 2 D3	DSR2D3	00$_H$	8	FFE6 140B$_H$	1(1H)	8
ATU4D	Timer Status Register D3	TSRD3	0000$_H$	16	FFE6 140C$_H$	1(1H)	8, 16
ATU4D	Timer Status Clear Register D3	TSCRD3	0000$_H$	16	FFE6 140E$_H$	1(1H)	8, 16
ATU4D	Timer Output Control Register D3	TOCRD3	00$_H$	8	FFE6 1410$_H$	1(1H)	8
ATU4D	Output Selection Register D3	OSELRD3	00$_H$	8	FFE6 1411$_H$	1(1H)	8
ATU4D	Output Value Register D3	ODRD3	00$_H$	8	FFE6 1412$_H$	1(1H)	8
ATU4D	Timer Input Capture Trigger Selection Register D3	TICTSELD3	01$_H$	8	FFE6 1414$_H$	1(1H)	8
ATU4D	Timer Offset Base Register D3	OSBRD3	0000 0000$_H$	32	FFE6 1418$_H$	1(1H)	32
ATU4D	Timer counter 1 D3	TCNT1D3	0000 0000$_H$	32	FFE6 1420$_H$	1(1H)	32
ATU4D	Timer counter 2 D3	TCNT2D3	0000 0000$_H$	32	FFE6 1424$_H$	1(1H)	32
ATU4D	Counter Upper-limit Setting Comparison Register 1 D3	CUCR1D3	FFFF FFFF$_H$	32	FFE6 1428$_H$	1(1H)	32
ATU4D	Counter Upper-limit Setting Comparison Register 2 D3	CUCR2D3	FFFF FFFF$_H$	32	FFE6 142C$_H$	1(1H)	32
ATU4D	Range Comparison Value Setting Register 1 D3	RCR1D3	00$_H$	8	FFE6 1430$_H$	1(1H)	8
ATU4D	Range Comparison Value Setting Register 2 D3	RCR2D3	00$_H$	8	FFE6 1434$_H$	1(1H)	8
ATU4D	Output Comparison Register 1 D30	OCR1D30	FFFF FFFF$_H$	32	FFE6 1440$_H$	1(1H)	32
ATU4D	Output Comparison Register 2 D30	OCR2D30	FFFF FFFF$_H$	32	FFE6 1448$_H$	1(1H)	32
ATU4D	Input Capture Register 1 D30	ICR1D30	0000 0000$_H$	32	FFE6 1450$_H$	1(1H)	32
ATU4D	Input Capture Register 2 D30	ICR2D30	0000 0000$_H$	32	FFE6 1454$_H$	1(1H)	32
ATU4D	Timer Down Counter D30	DCNTD30	0000 0000$_H$	32	FFE6 1458$_H$	1(1H)	32
ATU4D	Output Comparison Register 1 D31	OCR1D31	FFFF FFFF$_H$	32	FFE6 1460$_H$	1(1H)	32
ATU4D	Output Comparison Register 2 D31	OCR2D31	FFFF FFFF$_H$	32	FFE6 1468$_H$	1(1H)	32
ATU4D	Input Capture Register 1 D31	ICR1D31	0000 0000$_H$	32	FFE6 1470$_H$	1(1H)	32
ATU4D	Input Capture Register 2 D31	ICR2D31	0000 0000$_H$	32	FFE6 1474$_H$	1(1H)	32
ATU4D	Timer Down Counter D31	DCNTD31	0000 0000$_H$	32	FFE6 1478$_H$	1(1H)	32
ATU4D	Output Comparison Register 1 D32	OCR1D32	FFFF FFFF$_H$	32	FFE6 1480$_H$	1(1H)	32
ATU4D	Output Comparison Register 2 D32	OCR2D32	FFFF FFFF$_H$	32	FFE6 1488$_H$	1(1H)	32
ATU4D	Input Capture Register 1 D32	ICR1D32	0000 0000$_H$	32	FFE6 1490$_H$	1(1H)	32
ATU4D	Input Capture Register 2 D32	ICR2D32	0000 0000$_H$	32	FFE6 1494$_H$	1(1H)	32
ATU4D	Timer Down Counter D32	DCNTD32	0000 0000$_H$	32	FFE6 1498$_H$	1(1H)	32
ATU4D	Output Comparison Register 1 D33	OCR1D33	FFFF FFFF$_H$	32	FFE6 14A0$_H$	1(1H)	32
ATU4D	Output Comparison Register 2 D33	OCR2D33	FFFF FFFF$_H$	32	FFE6 14A8$_H$	1(1H)	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
ATU4D	Input Capture Register 1 D33	ICR1D33	0000 0000$_H$	32	FFE6 14B0$_H$	1(1H)	32
ATU4D	Input Capture Register 2 D33	ICR2D33	0000 0000$_H$	32	FFE6 14B4$_H$	1(1H)	32
ATU4D	Timer Down Counter D33	DCNTD33	0000 0000$_H$	32	FFE6 14B8$_H$	1(1H)	32
ATU4D	Timer Control Register D4	TCRD4	0000$_H$	16	FFE6 1500$_H$	1(1H)	8, 16
ATU4D	Down Counter Control Register D4	DCRD4	0000$_H$	16	FFE6 1502$_H$	1(1H)	8, 16
ATU4D	Timer I/O Control Register 1 D4	TIOR1D4	0000$_H$	16	FFE6 1504$_H$	1(1H)	8, 16
ATU4D	Timer I/O Control Register 2 D4	TIOR2D4	0000$_H$	16	FFE6 1506$_H$	1(1H)	8, 16
ATU4D	Down Count Start Register D4	DSTRD4	00$_H$	8	FFE6 1508$_H$	1(1H)	8
ATU4D	Down Count Status Clear Register D4	DSCRD4	00$_H$	8	FFE6 1509$_H$	1(1H)	8
ATU4D	Down Count Status Register 1D4	DSR1D4	00$_H$	8	FFE6 150A$_H$	1(1H)	8
ATU4D	Down Count Status Register 2D4	DSR2D4	00$_H$	8	FFE6 150B$_H$	1(1H)	8
ATU4D	Timer Status Register D4	TSRD4	0000$_H$	16	FFE6 150C$_H$	1(1H)	8, 16
ATU4D	Timer Status Clear Register D4	TSCRD4	0000$_H$	16	FFE6 150E$_H$	1(1H)	8, 16
ATU4D	Timer Output Control Register D4	TOCRD4	00$_H$	8	FFE6 1510$_H$	1(1H)	8
ATU4D	Output Selection Register D4	OSELRD4	00$_H$	8	FFE6 1511$_H$	1(1H)	8
ATU4D	Output Value Register D4	ODRD4	00$_H$	8	FFE6 1512$_H$	1(1H)	8
ATU4D	Timer Input Capture Trigger Selection Register D4	TICTSELD4	01$_H$	8	FFE6 1514$_H$	1(1H)	8
ATU4D	Timer Offset Base Register D4	OSBRD4	0000 0000$_H$	32	FFE6 1518$_H$	1(1H)	32
ATU4D	Timer counter 1 D4	TCNT1D4	0000 0000$_H$	32	FFE6 1520$_H$	1(1H)	32
ATU4D	Timer counter 2 D4	TCNT2D4	0000 0000$_H$	32	FFE6 1524$_H$	1(1H)	32
ATU4D	Counter Upper-limit Setting Comparison Register 1 D4	CUCR1D4	FFFF FFFF$_H$	32	FFE6 1528$_H$	1(1H)	32
ATU4D	Counter Upper-limit Setting Comparison Register 2 D4	CUCR2D4	FFFF FFFF$_H$	32	FFE6 152C$_H$	1(1H)	32
ATU4D	Range Comparison Value Setting Register 1 D4	RCR1D4	00$_H$	8	FFE6 1530$_H$	1(1H)	8
ATU4D	Range Comparison Value Setting Register 2 D4	RCR2D4	00$_H$	8	FFE6 1534$_H$	1(1H)	8
ATU4D	Output Comparison Register 1 D40	OCR1D40	FFFF FFFF$_H$	32	FFE6 1540$_H$	1(1H)	32
ATU4D	Output Comparison Register 2 D40	OCR2D40	FFFF FFFF$_H$	32	FFE6 1548$_H$	1(1H)	32
ATU4D	Input Capture Register 1 D40	ICR1D40	0000 0000$_H$	32	FFE6 1550$_H$	1(1H)	32
ATU4D	Input Capture Register 2 D40	ICR2D40	0000 0000$_H$	32	FFE6 1554$_H$	1(1H)	32
ATU4D	Timer Down Counter D40	DCNTD40	0000 0000$_H$	32	FFE6 1558$_H$	1(1H)	32
ATU4D	Output Comparison Register 1D 41	OCR1D41	FFFF FFFF$_H$	32	FFE6 1560$_H$	1(1H)	32
ATU4D	Output Comparison Register 2 D41	OCR2D41	FFFF FFFF$_H$	32	FFE6 1568$_H$	1(1H)	32
ATU4D	Input Capture Register 1 D41	ICR1D41	0000 0000$_H$	32	FFE6 1570$_H$	1(1H)	32
ATU4D	Input Capture Register 2 D41	ICR2D41	0000 0000$_H$	32	FFE6 1574$_H$	1(1H)	32
ATU4D	Timer Down Counter D41	DCNTD41	0000 0000$_H$	32	FFE6 1578$_H$	1(1H)	32
ATU4D	Output Comparison Register 1 D42	OCR1D42	FFFF FFFF$_H$	32	FFE6 1580$_H$	1(1H)	32
ATU4D	Output Comparison Register 2 D42	OCR2D42	FFFF FFFF$_H$	32	FFE6 1588$_H$	1(1H)	32
ATU4D	Input Capture Register 1 D42	ICR1D42	0000 0000$_H$	32	FFE6 1590$_H$	1(1H)	32
ATU4D	Input Capture Register 2 D42	ICR2D42	0000 0000$_H$	32	FFE6 1594$_H$	1(1H)	32
ATU4D	Timer Down Counter D42	DCNTD42	0000 0000$_H$	32	FFE6 1598$_H$	1(1H)	32
ATU4D	Output Comparison Register 1 D43	OCR1D43	FFFF FFFF$_H$	32	FFE6 15A0$_H$	1(1H)	32
ATU4D	Output Comparison Register 2 D43	OCR2D43	FFFF FFFF$_H$	32	FFE6 15A8$_H$	1(1H)	32
ATU4D	Input Capture Register 1 D43	ICR1D43	0000 0000$_H$	32	FFE6 15B0$_H$	1(1H)	32
ATU4D	Input Capture Register 2 D43	ICR2D43	0000 0000$_H$	32	FFE6 15B4$_H$	1(1H)	32
ATU4D	Timer Down Counter D43	DCNTD43	0000 0000$_H$	32	FFE6 15B8$_H$	1(1H)	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
ATU4D	Timer Control Register D5	TCRD5	0000_H	16	FFE6 1600_H	1(1H)	8, 16
ATU4D	Down Counter Control Register D5	DCRD5	0000_H	16	FFE6 1602_H	1(1H)	8, 16
ATU4D	Timer I/O Control Register 1 D5	TIOR1D5	0000_H	16	FFE6 1604_H	1(1H)	8, 16
ATU4D	Timer I/O Control Register 2 D5	TIOR2D5	0000_H	16	FFE6 1606_H	1(1H)	8, 16
ATU4D	Down Count Start Register D5	DSTRD5	00_H	8	FFE6 1608_H	1(1H)	8
ATU4D	Down Count Status Clear Register D5	DSCRD5	00_H	8	FFE6 1609_H	1(1H)	8
ATU4D	Down Count Status Register 1 D5	DSR1D5	00_H	8	FFE6 $160A_H$	1(1H)	8
ATU4D	Down Count Status Register 2 D5	DSR2D5	00_H	8	FFE6 $160B_H$	1(1H)	8
ATU4D	Timer Status Register D5	TSRD5	0000_H	16	FFE6 $160C_H$	1(1H)	8, 16
ATU4D	Timer Status Clear Register D5	TSCRD5	0000_H	16	FFE6 $160E_H$	1(1H)	8, 16
ATU4D	Timer Output Control Register D5	TOCRD5	00_H	8	FFE6 1610_H	1(1H)	8
ATU4D	Output Selection Register D5	OSELRD5	00_H	8	FFE6 1611_H	1(1H)	8
ATU4D	Output Value Register D5	ODRD5	00_H	8	FFE6 1612_H	1(1H)	8
ATU4D	Timer Input Capture Trigger Selection Register D5	TICTSELD5	01_H	8	FFE6 1614_H	1(1H)	8
ATU4D	Timer Offset Base Register D5	OSBRD5	0000 0000_H	32	FFE6 1618_H	1(1H)	32
ATU4D	Timer counter 1 D5	TCNT1D5	0000 0000_H	32	FFE6 1620_H	1(1H)	32
ATU4D	Timer counter 2 D5	TCNT2D5	0000 0000_H	32	FFE6 1624_H	1(1H)	32
ATU4D	Counter Upper-limit Setting Comparison Register 1 D5	CUCR1D5	FFFF $FFFF_H$	32	FFE6 1628_H	1(1H)	32
ATU4D	Counter Upper-limit Setting Comparison Register 2 D5	CUCR2D5	FFFF $FFFF_H$	32	FFE6 $162C_H$	1(1H)	32
ATU4D	Range Comparison Value Setting Register 1 D5	RCR1D5	00_H	8	FFE6 1630_H	1(1H)	8
ATU4D	Range Comparison Value Setting Register 2 D5	RCR2D5	00_H	8	FFE6 1634_H	1(1H)	8
ATU4D	Output Comparison Register 1 D50	OCR1D50	FFFF $FFFF_H$	32	FFE6 1640_H	1(1H)	32
ATU4D	Output Comparison Register 2 D50	OCR2D50	FFFF $FFFF_H$	32	FFE6 1648_H	1(1H)	32
ATU4D	Input Capture Register 1 D50	ICR1D50	0000 0000_H	32	FFE6 1650_H	1(1H)	32
ATU4D	Input Capture Register 2 D50	ICR2D50	0000 0000_H	32	FFE6 1654_H	1(1H)	32
ATU4D	Timer Down Counter D50	DCNTD50	0000 0000_H	32	FFE6 1658_H	1(1H)	32
ATU4D	Output Comparison Register 1 D51	OCR1D51	FFFF $FFFF_H$	32	FFE6 1660_H	1(1H)	32
ATU4D	Output Comparison Register 2 D51	OCR2D51	FFFF $FFFF_H$	32	FFE6 1668_H	1(1H)	32
ATU4D	Input Capture Register 1 D51	ICR1D51	0000 0000_H	32	FFE6 1670_H	1(1H)	32
ATU4D	Input Capture Register 2 D51	ICR2D51	0000 0000_H	32	FFE6 1674_H	1(1H)	32
ATU4D	Timer Down Counter D51	DCNTD51	0000 0000_H	32	FFE6 1678_H	1(1H)	32
ATU4D	Output Comparison Register 1 D52	OCR1D52	FFFF $FFFF_H$	32	FFE6 1680_H	1(1H)	32
ATU4D	Output Comparison Register 2 D52	OCR2D52	FFFF $FFFF_H$	32	FFE6 1688_H	1(1H)	32
ATU4D	Input Capture Register 1 D52	ICR1D52	0000 0000_H	32	FFE6 1690_H	1(1H)	32
ATU4D	Input Capture Register 2 D52	ICR2D52	0000 0000_H	32	FFE6 1694_H	1(1H)	32
ATU4D	Timer Down Counter D52	DCNTD52	0000 0000_H	32	FFE6 1698_H	1(1H)	32
ATU4D	Output Comparison Register 1 D53	OCR1D53	FFFF $FFFF_H$	32	FFE6 $16A0_H$	1(1H)	32
ATU4D	Output Comparison Register 2 D53	OCR2D53	FFFF $FFFF_H$	32	FFE6 $16A8_H$	1(1H)	32
ATU4D	Input Capture Register 1 D53	ICR1D53	0000 0000_H	32	FFE6 $16B0_H$	1(1H)	32
ATU4D	Input Capture Register 2 D53	ICR2D53	0000 0000_H	32	FFE6 $16B4_H$	1(1H)	32
ATU4D	Timer Down Counter D53	DCNTD53	0000 0000_H	32	FFE6 $16B8_H$	1(1H)	32
ATU4D	Timer Control Register D6	TCRD6	0000_H	16	FFE6 1700_H	1(1H)	8, 16
ATU4D	Down Counter Control Register D6	DCRD6	0000_H	16	FFE6 1702_H	1(1H)	8, 16
ATU4D	Timer I/O Control Register 1 D6	TIOR1D6	0000_H	16	FFE6 1704_H	1(1H)	8, 16

(81/201)

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
ATU4D	Timer I/O Control Register 2 D6	TIOR2D6	0000_H	16	FFE6 1706_H	1(1H)	8, 16
ATU4D	Down Count Start Register D6	DSTRD6	00_H	8	FFE6 1708_H	1(1H)	8
ATU4D	Down Count Status Clear Register D6	DSCRD6	00_H	8	FFE6 1709_H	1(1H)	8
ATU4D	Down Count Status Register 1 D6	DSR1D6	00_H	8	FFE6 $170A_H$	1(1H)	8
ATU4D	Down Count Status Register 2 D6	DSR2D6	00_H	8	FFE6 $170B_H$	1(1H)	8
ATU4D	Timer Status Register D6	TSRD6	0000_H	16	FFE6 $170C_H$	1(1H)	8, 16
ATU4D	Timer Status Clear Register D6	TSCRD6	0000_H	16	FFE6 $170E_H$	1(1H)	8, 16
ATU4D	Timer Output Control Register D6	TOCRD6	00_H	8	FFE6 1710_H	1(1H)	8
ATU4D	Output Selection Register D6	OSELRD6	00_H	8	FFE6 1711_H	1(1H)	8
ATU4D	Output Value Register D6	ODRD6	00_H	8	FFE6 1712_H	1(1H)	8
ATU4D	Timer Input Capture Trigger Selection Register D6	TICTSELD6	01_H	8	FFE6 1714_H	1(1H)	8
ATU4D	Timer Offset Base Register D6	OSBRD6	0000 0000_H	32	FFE6 1718_H	1(1H)	32
ATU4D	Timer counter 1 D6	TCNT1D6	0000 0000_H	32	FFE6 1720_H	1(1H)	32
ATU4D	Timer counter 2 D6	TCNT2D6	0000 0000_H	32	FFE6 1724_H	1(1H)	32
ATU4D	Counter Upper-limit Setting Comparison Register 1 D6	CUCR1D6	FFFF $FFFF_H$	32	FFE6 1728_H	1(1H)	32
ATU4D	Counter Upper-limit Setting Comparison Register 2 D6	CUCR2D6	FFFF $FFFF_H$	32	FFE6 $172C_H$	1(1H)	32
ATU4D	Range Comparison Value Setting Register 1 D6	RCR1D6	00_H	8	FFE6 1730_H	1(1H)	8
ATU4D	Range Comparison Value Setting Register 2 D6	RCR2D6	00_H	8	FFE6 1734_H	1(1H)	8
ATU4D	Output Comparison Register 1 D60	OCR1D60	FFFF $FFFF_H$	32	FFE6 1740_H	1(1H)	32
ATU4D	Output Comparison Register 2 D60	OCR2D60	FFFF $FFFF_H$	32	FFE6 1748_H	1(1H)	32
ATU4D	Input Capture Register 1 D60	ICR1D60	0000 0000_H	32	FFE6 1750_H	1(1H)	32
ATU4D	Input Capture Register 2 D60	ICR2D60	0000 0000_H	32	FFE6 1754_H	1(1H)	32
ATU4D	Timer Down Counter D60	DCNTD60	0000 0000_H	32	FFE6 1758_H	1(1H)	32
ATU4D	Output Comparison Register 1 D61	OCR1D61	FFFF $FFFF_H$	32	FFE6 1760_H	1(1H)	32
ATU4D	Output Comparison Register 2 D61	OCR2D61	FFFF $FFFF_H$	32	FFE6 1768_H	1(1H)	32
ATU4D	Input Capture Register 1 D61	ICR1D61	0000 0000_H	32	FFE6 1770_H	1(1H)	32
ATU4D	Input Capture Register 2 D61	ICR2D61	0000 0000_H	32	FFE6 1774_H	1(1H)	32
ATU4D	Timer Down Counter D61	DCNTD61	0000 0000_H	32	FFE6 1778_H	1(1H)	32
ATU4D	Output Comparison Register 1 D62	OCR1D62	FFFF $FFFF_H$	32	FFE6 1780_H	1(1H)	32
ATU4D	Output Comparison Register 2 D62	OCR2D62	FFFF $FFFF_H$	32	FFE6 1788_H	1(1H)	32
ATU4D	Input Capture Register 1 D62	ICR1D62	0000 0000_H	32	FFE6 1790_H	1(1H)	32
ATU4D	Input Capture Register 2 D62	ICR2D62	0000 0000_H	32	FFE6 1794_H	1(1H)	32
ATU4D	Timer Down Counter D62	DCNTD62	0000 0000_H	32	FFE6 1798_H	1(1H)	32
ATU4D	Output Comparison Register 1 D63	OCR1D63	FFFF $FFFF_H$	32	FFE6 $17A0_H$	1(1H)	32
ATU4D	Output Comparison Register 2 D63	OCR2D63	FFFF $FFFF_H$	32	FFE6 $17A8_H$	1(1H)	32
ATU4D	Input Capture Register 1 D63	ICR1D63	0000 0000_H	32	FFE6 $17B0_H$	1(1H)	32
ATU4D	Input Capture Register 2 D63	ICR2D63	0000 0000_H	32	FFE6 $17B4_H$	1(1H)	32
ATU4D	Timer Down Counter D63	DCNTD63	0000 0000_H	32	FFE6 $17B8_H$	1(1H)	32
ATU4D	Timer Control Register D7	TCRD7	0000_H	16	FFE6 1800_H	1(1H)	8, 16
ATU4D	Down Counter Control Register D7	DCRD7	0000_H	16	FFE6 1802_H	1(1H)	8, 16
ATU4D	Timer I/O Control Register 1 D7	TIOR1D7	0000_H	16	FFE6 1804_H	1(1H)	8, 16
ATU4D	Timer I/O Control Register 2 D7	TIOR2D7	0000_H	16	FFE6 1806_H	1(1H)	8, 16
ATU4D	Down Count Start Register D7	DSTRD7	00_H	8	FFE6 1808_H	1(1H)	8
ATU4D	Down Count Status Clear Register D7	DSCRD7	00_H	8	FFE6 1809_H	1(1H)	8

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
ATU4D	Down Count Status Register 1 D7	DSR1D7	00_H	8	FFE6 180A_H	1(1H)	8
ATU4D	Down Count Status Register 2 D7	DSR2D7	00_H	8	FFE6 180B_H	1(1H)	8
ATU4D	Timer Status Register D7	TSRD7	0000_H	16	FFE6 180C_H	1(1H)	8, 16
ATU4D	Timer Status Clear Register D7	TSCRD7	0000_H	16	FFE6 180E_H	1(1H)	8, 16
ATU4D	Timer Output Control Register D7	TOCRD7	00_H	8	FFE6 1810_H	1(1H)	8
ATU4D	Output Selection Register D7	OSELRD7	00_H	8	FFE6 1811_H	1(1H)	8
ATU4D	Output Value Register D7	ODRD7	00_H	8	FFE6 1812_H	1(1H)	8
ATU4D	Timer Input Capture Trigger Selection Register D7	TICTSELD7	01_H	8	FFE6 1814_H	1(1H)	8
ATU4D	Timer Offset Base Register D7	OSBRD7	$0000\ 0000_H$	32	FFE6 1818_H	1(1H)	32
ATU4D	Timer counter 1 D7	TCNT1D7	$0000\ 0000_H$	32	FFE6 1820_H	1(1H)	32
ATU4D	Timer counter 2 D7	TCNT2D7	$0000\ 0000_H$	32	FFE6 1824_H	1(1H)	32
ATU4D	Counter Upper-limit Setting Comparison Register 1 D7	CUCR1D7	$FFFF\ FFFF_H$	32	FFE6 1828_H	1(1H)	32
ATU4D	Counter Upper-limit Setting Comparison Register 2 D7	CUCR2D7	$FFFF\ FFFF_H$	32	FFE6 182C_H	1(1H)	32
ATU4D	Range Comparison Value Setting Register 1 D7	RCR1D7	00_H	8	FFE6 1830_H	1(1H)	8
ATU4D	Range Comparison Value Setting Register 2 D7	RCR2D7	00_H	8	FFE6 1834_H	1(1H)	8
ATU4D	Output Comparison Register 1 D70	OCR1D70	$FFFF\ FFFF_H$	32	FFE6 1840_H	1(1H)	32
ATU4D	Output Comparison Register 2 D70	OCR2D70	$FFFF\ FFFF_H$	32	FFE6 1848_H	1(1H)	32
ATU4D	Input Capture Register 1 D70	ICR1D70	$0000\ 0000_H$	32	FFE6 1850_H	1(1H)	32
ATU4D	Input Capture Register 2 D70	ICR2D70	$0000\ 0000_H$	32	FFE6 1854_H	1(1H)	32
ATU4D	Timer Down Counter D70	DCNTD70	$0000\ 0000_H$	32	FFE6 1858_H	1(1H)	32
ATU4D	Output Comparison Register 1 D71	OCR1D71	$FFFF\ FFFF_H$	32	FFE6 1860_H	1(1H)	32
ATU4D	Output Comparison Register 2 D71	OCR2D71	$FFFF\ FFFF_H$	32	FFE6 1868_H	1(1H)	32
ATU4D	Input Capture Register 1 D71	ICR1D71	$0000\ 0000_H$	32	FFE6 1870_H	1(1H)	32
ATU4D	Input Capture Register 2 D71	ICR2D71	$0000\ 0000_H$	32	FFE6 1874_H	1(1H)	32
ATU4D	Timer Down Counter D71	DCNTD71	$0000\ 0000_H$	32	FFE6 1878_H	1(1H)	32
ATU4D	Output Comparison Register 1 D72	OCR1D72	$FFFF\ FFFF_H$	32	FFE6 1880_H	1(1H)	32
ATU4D	Output Comparison Register 2 D72	OCR2D72	$FFFF\ FFFF_H$	32	FFE6 1888_H	1(1H)	32
ATU4D	Input Capture Register 1 D72	ICR1D72	$0000\ 0000_H$	32	FFE6 1890_H	1(1H)	32
ATU4D	Input Capture Register 2 D72	ICR2D72	$0000\ 0000_H$	32	FFE6 1894_H	1(1H)	32
ATU4D	Timer Down Counter D72	DCNTD72	$0000\ 0000_H$	32	FFE6 1898_H	1(1H)	32
ATU4D	Output Comparison Register 1 D73	OCR1D73	$FFFF\ FFFF_H$	32	FFE6 18A0_H	1(1H)	32
ATU4D	Output Comparison Register 2 D73	OCR2D73	$FFFF\ FFFF_H$	32	FFE6 18A8_H	1(1H)	32
ATU4D	Input Capture Register 1 D73	ICR1D73	$0000\ 0000_H$	32	FFE6 18B0_H	1(1H)	32
ATU4D	Input Capture Register 2 D73	ICR2D73	$0000\ 0000_H$	32	FFE6 18B4_H	1(1H)	32
ATU4D	Timer Down Counter D73	DCNTD73	$0000\ 0000_H$	32	FFE6 18B8_H	1(1H)	32
ATU4D	Timer Control Register D8	TCRD8	0000_H	16	FFE6 1900_H	1(1H)	8, 16
ATU4D	Down Counter Control Register D8	DCRD8	0000_H	16	FFE6 1902_H	1(1H)	8, 16
ATU4D	Timer I/O Control Register 1 D8	TIOR1D8	0000_H	16	FFE6 1904_H	1(1H)	8, 16
ATU4D	Timer I/O Control Register 2 D8	TIOR2D8	0000_H	16	FFE6 1906_H	1(1H)	8, 16
ATU4D	Down Count Start Register D8	DSTRD8	00_H	8	FFE6 1908_H	1(1H)	8
ATU4D	Down Count Status Clear Register D8	DSCRD8	00_H	8	FFE6 1909_H	1(1H)	8
ATU4D	Down Count Status Register 1 D8	DSR1D8	00_H	8	FFE6 190A_H	1(1H)	8
ATU4D	Down Count Status Register 2 D8	DSR2D8	00_H	8	FFE6 190B_H	1(1H)	8
ATU4D	Timer Status Register D8	TSRD8	0000_H	16	FFE6 190C_H	1(1H)	8, 16

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
ATU4D	Timer Status Clear Register D8	TSCRD8	0000_H	16	FFE6 190E$_H$	1(1H)	8, 16
ATU4D	Timer Output Control Register D8	TOCRD8	00_H	8	FFE6 1910$_H$	1(1H)	8
ATU4D	Output Selection Register D8	OSELRD8	00_H	8	FFE6 1911$_H$	1(1H)	8
ATU4D	Output Value Register D8	ODRD8	00_H	8	FFE6 1912$_H$	1(1H)	8
ATU4D	Timer Input Capture Trigger Selection Register D8	TICTSELD8	01_H	8	FFE6 1914$_H$	1(1H)	8
ATU4D	Timer Offset Base Register D8	OSBRD8	0000 0000_H	32	FFE6 1918$_H$	1(1H)	32
ATU4D	Timer counter 1 D8	TCNT1D8	0000 0000_H	32	FFE6 1920$_H$	1(1H)	32
ATU4D	Timer counter 2 D8	TCNT2D8	0000 0000_H	32	FFE6 1924$_H$	1(1H)	32
ATU4D	Counter Upper-limit Setting Comparison Register 1 D8	CUCR1D8	FFFF $FFFF_H$	32	FFE6 1928$_H$	1(1H)	32
ATU4D	Counter Upper-limit Setting Comparison Register 2 D8	CUCR2D8	FFFF $FFFF_H$	32	FFE6 192C$_H$	1(1H)	32
ATU4D	Range Comparison Value Setting Register 1 D8	RCR1D8	00_H	8	FFE6 1930$_H$	1(1H)	8
ATU4D	Range Comparison Value Setting Register 2 D8	RCR2D8	00_H	8	FFE6 1934$_H$	1(1H)	8
ATU4D	Output Comparison Register 1 D80	OCR1D80	FFFF $FFFF_H$	32	FFE6 1940$_H$	1(1H)	32
ATU4D	Output Comparison Register 2 D80	OCR2D80	FFFF $FFFF_H$	32	FFE6 1948$_H$	1(1H)	32
ATU4D	Input Capture Register 1 D80	ICR1D80	0000 0000_H	32	FFE6 1950$_H$	1(1H)	32
ATU4D	Input Capture Register 2 D80	ICR2D80	0000 0000_H	32	FFE6 1954$_H$	1(1H)	32
ATU4D	Timer Down Counter D80	DCNTD80	0000 0000_H	32	FFE6 1958$_H$	1(1H)	32
ATU4D	Output Comparison Register 1 D81	OCR1D81	FFFF $FFFF_H$	32	FFE6 1960$_H$	1(1H)	32
ATU4D	Output Comparison Register 2 D81	OCR2D81	FFFF $FFFF_H$	32	FFE6 1968$_H$	1(1H)	32
ATU4D	Input Capture Register 1 D81	ICR1D81	0000 0000_H	32	FFE6 1970$_H$	1(1H)	32
ATU4D	Input Capture Register 2 D81	ICR2D81	0000 0000_H	32	FFE6 1974$_H$	1(1H)	32
ATU4D	Timer Down Counter D81	DCNTD81	0000 0000_H	32	FFE6 1978$_H$	1(1H)	32
ATU4D	Output Comparison Register D82	OCR1D82	FFFF $FFFF_H$	32	FFE6 1980$_H$	1(1H)	32
ATU4D	Output Comparison Register D82	OCR2D82	FFFF $FFFF_H$	32	FFE6 1988$_H$	1(1H)	32
ATU4D	Input Capture Register 1 D82	ICR1D82	0000 0000_H	32	FFE6 1990$_H$	1(1H)	32
ATU4D	Input Capture Register 2 D82	ICR2D82	0000 0000_H	32	FFE6 1994$_H$	1(1H)	32
ATU4D	Timer Down Counter D82	DCNTD82	0000 0000_H	32	FFE6 1998$_H$	1(1H)	32
ATU4D	Output Comparison Register 1 D83	OCR1D83	FFFF $FFFF_H$	32	FFE6 19A0$_H$	1(1H)	32
ATU4D	Output Comparison Register 2 D83	OCR2D83	FFFF $FFFF_H$	32	FFE6 19A8$_H$	1(1H)	32
ATU4D	Input Capture Register 1 D83	ICR1D83	0000 0000_H	32	FFE6 19B0$_H$	1(1H)	32
ATU4D	Input Capture Register 2 D83	ICR2D83	0000 0000_H	32	FFE6 19B4$_H$	1(1H)	32
ATU4D	Timer Down Counter D83	DCNTD83	0000 0000_H	32	FFE6 19B8$_H$	1(1H)	32
ATU4D	Timer Control Register D9	TCRD9	0000_H	16	FFE6 1A00$_H$	1(1H)	8, 16
ATU4D	Down Counter Control Register D9	DCRD9	0000_H	16	FFE6 1A02$_H$	1(1H)	8, 16
ATU4D	Timer I/O Control Register 1 D9	TIOR1D9	0000_H	16	FFE6 1A04$_H$	1(1H)	8, 16
ATU4D	Timer I/O Control Register 2 D9	TIOR2D9	0000_H	16	FFE6 1A06$_H$	1(1H)	8, 16
ATU4D	Down Count Start Register D9	DSTRD9	00_H	8	FFE6 1A08$_H$	1(1H)	8
ATU4D	Down Count Status Clear Register D9	DSCRD9	00_H	8	FFE6 1A09$_H$	1(1H)	8
ATU4D	Down Count Status Register 1 D9	DSR1D9	00_H	8	FFE6 1A0A$_H$	1(1H)	8
ATU4D	Down Count Status Register 2 D9	DSR2D9	00_H	8	FFE6 1A0B$_H$	1(1H)	8
ATU4D	Timer Status Register D9	TSRD9	0000_H	16	FFE6 1A0C$_H$	1(1H)	8, 16
ATU4D	Timer Status Clear Register D9	TSCRD9	0000_H	16	FFE6 1A0E$_H$	1(1H)	8, 16
ATU4D	Timer Output Control Register D9	TOCRD9	00_H	8	FFE6 1A10$_H$	1(1H)	8
ATU4D	Output Selection Register D9	OSELRD9	00_H	8	FFE6 1A11$_H$	1(1H)	8

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
ATU4D	Output Value Register D9	ODRD9	00_H	8	FFE6 $1A12_H$	1(1H)	8
ATU4D	Timer Input Capture Trigger Selection Register D9	TICTSELD9	01_H	8	FFE6 $1A14_H$	1(1H)	8
ATU4D	Timer Offset Base Register D9	OSBRD9	0000 0000_H	32	FFE6 $1A18_H$	1(1H)	32
ATU4D	Timer Counter 1 D9	TCNT1D9	0000 0000_H	32	FFE6 $1A20_H$	1(1H)	32
ATU4D	Timer Counter 2 D9	TCNT2D9	0000 0000_H	32	FFE6 $1A24_H$	1(1H)	32
ATU4D	Counter Upper-limit Setting Comparison Register 1 D9	CUCR1D9	FFFF $FFFF_H$	32	FFE6 $1A28_H$	1(1H)	32
ATU4D	Counter Upper-limit Setting Comparison Register 2 D9	CUCR2D9	FFFF $FFFF_H$	32	FFE6 $1A2C_H$	1(1H)	32
ATU4D	Range Comparison Value Setting Register 1 D9	RCR1D9	00_H	8	FFE6 $1A30_H$	1(1H)	8
ATU4D	Range Comparison Value Setting Register 2 D9	RCR2D9	00_H	8	FFE6 $1A34_H$	1(1H)	8
ATU4D	Output Comparison Register 1 D90	OCR1D90	FFFF $FFFF_H$	32	FFE6 $1A40_H$	1(1H)	32
ATU4D	Output Comparison Register 2 D90	OCR2D90	FFFF $FFFF_H$	32	FFE6 $1A48_H$	1(1H)	32
ATU4D	Input Capture Register 1 D90	ICR1D90	0000 0000_H	32	FFE6 $1A50_H$	1(1H)	32
ATU4D	Input Capture Register 2 D90	ICR2D90	0000 0000_H	32	FFE6 $1A54_H$	1(1H)	32
ATU4D	Timer Down Counter D90	DCNTD90	0000 0000_H	32	FFE6 $1A58_H$	1(1H)	32
ATU4D	Output Comparison Register 1 D91	OCR1D91	FFFF $FFFF_H$	32	FFE6 $1A60_H$	1(1H)	32
ATU4D	Output Comparison Register 2 D91	OCR2D91	FFFF $FFFF_H$	32	FFE6 $1A68_H$	1(1H)	32
ATU4D	Input Capture Register 1 D91	ICR1D91	0000 0000_H	32	FFE6 $1A70_H$	1(1H)	32
ATU4D	Input Capture Register 2 D91	ICR2D91	0000 0000_H	32	FFE6 $1A74_H$	1(1H)	32
ATU4D	Timer Down Counter D91	DCNTD91	0000 0000_H	32	FFE6 $1A78_H$	1(1H)	32
ATU4D	Output Comparison Register 1 D92	OCR1D92	FFFF $FFFF_H$	32	FFE6 $1A80_H$	1(1H)	32
ATU4D	Output Comparison Register 2 D92	OCR2D92	FFFF $FFFF_H$	32	FFE6 $1A88_H$	1(1H)	32
ATU4D	Input Capture Register 1 D92	ICR1D92	0000 0000_H	32	FFE6 $1A90_H$	1(1H)	32
ATU4D	Input Capture Register 2 D92	ICR2D92	0000 0000_H	32	FFE6 $1A94_H$	1(1H)	32
ATU4D	Timer Down Counter D92	DCNTD92	0000 0000_H	32	FFE6 $1A98_H$	1(1H)	32
ATU4D	Output Comparison Register 1 D93	OCR1D93	FFFF $FFFF_H$	32	FFE6 $1AA0_H$	1(1H)	32
ATU4D	Output Comparison Register 2 D93	OCR2D93	FFFF $FFFF_H$	32	FFE6 $1AA8_H$	1(1H)	32
ATU4D	Input Capture Register 1 D93	ICR1D93	0000 0000_H	32	FFE6 $1AB0_H$	1(1H)	32
ATU4D	Input Capture Register 2 D93	ICR2D93	0000 0000_H	32	FFE6 $1AB4_H$	1(1H)	32
ATU4D	Timer Down Counter D93	DCNTD93	0000 0000_H	32	FFE6 $1AB8_H$	1(1H)	32
ATU4E	Timer Start Register E	TSTRE	00_H	8	FFE6 2000_H	1(1H)	8
ATU4E	Sub-block Start Register E0	SSTRE0	00_H	8	FFE6 2100_H	1(1H)	8
ATU4E	Prescaler Register E0	PSCRE0	00_H	8	FFE6 2104_H	1(1H)	8
ATU4E	Timer Control Register E0	TCRE0	00_H	8	FFE6 2108_H	1(1H)	8
ATU4E	Reload Control Register E0	RLDCRE0	00_H	8	FFE6 $210A_H$	1(1H)	8
ATU4E	Output Shutoff Control Register E0	POECRE0	0000_H	16	FFE6 $210C_H$	1(1H)	16
ATU4E	Output Shutoff Level Setting Register E0	SOLVLE0	00_H	8	FFE6 $210E_H$	1(1H)	8
ATU4E	Timer Status Register E0	TSRE0	0000_H	16	FFE6 2110_H	1(1H)	8, 16
ATU4E	Timer Status Clear Register E0	TSCRE0	0000_H	16	FFE6 2112_H	1(1H)	8, 16
ATU4E	Timer Output Control Register E0	TOCRE0	00_H	8	FFE6 2114_H	1(1H)	8
ATU4E	Timer Interrupt Enable Register E0	TIERE0	0000_H	16	FFE6 2116_H	1(1H)	8, 16
ATU4E	Prescaler Channel Register E00	PSCCRE00	00_H	8	FFE6 2118_H	1(1H)	8
ATU4E	Prescaler Channel Register E01	PSCCRE01	00_H	8	FFE6 2119_H	1(1H)	8
ATU4E	Prescaler Channel Register E02	PSCCRE02	00_H	8	FFE6 $211A_H$	1(1H)	8
ATU4E	Prescaler Channel Register E03	PSCCRE03	00_H	8	FFE6 $211B_H$	1(1H)	8

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
ATU4E	Timer counter E00	TCNTE00	0000 0100_H	32	FFE6 2124_H	1(1H)	32
ATU4E	Cycle Register E00	CYLRE00	FFFF $FF00_H$	32	FFE6 2128_H	1(1H)	32
ATU4E	Duty Register E00	DTRE00	FFFF $FF00_H$	32	FFE6 $212C_H$	1(1H)	32
ATU4E	Cycle Reload Register E00	CRLDE00	FFFF $FF00_H$	32	FFE6 2130_H	1(1H)	32
ATU4E	Duty Reload Register E00	DRLDE00	FFFF $FF00_H$	32	FFE6 2134_H	1(1H)	32
ATU4E	Timer counter E01	TCNTE01	0000 0100_H	32	FFE6 2144_H	1(1H)	32
ATU4E	Cycle Register E01	CYLRE01	FFFF $FF00_H$	32	FFE6 2148_H	1(1H)	32
ATU4E	Duty Register E01	DTRE01	FFFF $FF00_H$	32	FFE6 $214C_H$	1(1H)	32
ATU4E	Cycle Reload Register E01	CRLDE01	FFFF $FF00_H$	32	FFE6 2150_H	1(1H)	32
ATU4E	Duty Reload Register E01	DRLDE01	FFFF $FF00_H$	32	FFE6 2154_H	1(1H)	32
ATU4E	Timer counter E02	TCNTE02	0000 0100_H	32	FFE6 2164_H	1(1H)	32
ATU4E	Cycle Register E02	CYLRE02	FFFF $FF00_H$	32	FFE6 2168_H	1(1H)	32
ATU4E	Duty Register E02	DTRE02	FFFF $FF00_H$	32	FFE6 $216C_H$	1(1H)	32
ATU4E	Cycle Reload Register E02	CRLDE02	FFFF $FF00_H$	32	FFE6 2170_H	1(1H)	32
ATU4E	Duty Reload Register E02	DRLDE02	FFFF $FF00_H$	32	FFE6 2174_H	1(1H)	32
ATU4E	Timer counter E03	TCNTE03	0000 0100_H	32	FFE6 2184_H	1(1H)	32
ATU4E	Cycle Register E03	CYLRE03	FFFF $FF00_H$	32	FFE6 2188_H	1(1H)	32
ATU4E	Duty Register E03	DTRE03	FFFF $FF00_H$	32	FFE6 $218C_H$	1(1H)	32
ATU4E	Cycle Reload Register E03	CRLDE03	FFFF $FF00_H$	32	FFE6 2190_H	1(1H)	32
ATU4E	Duty Reload Register E03	DRLDE03	FFFF $FF00_H$	32	FFE6 2194_H	1(1H)	32
ATU4E	Sub-block Start Register E1	SSTRE1	00_H	8	FFE6 2200_H	1(1H)	8
ATU4E	Prescaler Register E1	PSCRE1	00_H	8	FFE6 2204_H	1(1H)	8
ATU4E	Timer Control Register E1	TCRE1	00_H	8	FFE6 2208_H	1(1H)	8
ATU4E	Reload Control Register E1	RLDCRE1	00_H	8	FFE6 $220A_H$	1(1H)	8
ATU4E	Output Shutoff Control Register E1	POECRE1	0000_H	16	FFE6 $220C_H$	1(1H)	16
ATU4E	Output Shutoff Level Setting Register E1	SOLVLE1	00_H	8	FFE6 $220E_H$	1(1H)	8
ATU4E	Timer Status Register E1	TSRE1	0000_H	16	FFE6 2210_H	1(1H)	8, 16
ATU4E	Timer Status Clear Register E1	TSCRE1	0000_H	16	FFE6 2212_H	1(1H)	8, 16
ATU4E	Timer Output Control Register E1	TOCRE1	00_H	8	FFE6 2214_H	1(1H)	8
ATU4E	Timer Interrupt Enable Register E1	TIERE1	0000_H	16	FFE6 2216_H	1(1H)	8, 16
ATU4E	Prescaler Channel Register E10	PSCCRE10	00_H	8	FFE6 2218_H	1(1H)	8
ATU4E	Prescaler Channel Register E11	PSCCRE11	00_H	8	FFE6 2219_H	1(1H)	8
ATU4E	Prescaler Channel Register E12	PSCCRE12	00_H	8	FFE6 $221A_H$	1(1H)	8
ATU4E	Prescaler Channel Register E13	PSCCRE13	00_H	8	FFE6 $221B_H$	1(1H)	8
ATU4E	Timer counter E10	TCNTE10	0000 0100_H	32	FFE6 2224_H	1(1H)	32
ATU4E	Cycle Register E10	CYLRE10	FFFF $FF00_H$	32	FFE6 2228_H	1(1H)	32
ATU4E	Duty Register E10	DTRE10	FFFF $FF00_H$	32	FFE6 $222C_H$	1(1H)	32
ATU4E	Cycle Reload Register E10	CRLDE10	FFFF $FF00_H$	32	FFE6 2230_H	1(1H)	32
ATU4E	Duty Reload Register E10	DRLDE10	FFFF $FF00_H$	32	FFE6 2234_H	1(1H)	32
ATU4E	Timer counter E11	TCNTE11	0000 0100_H	32	FFE6 2244_H	1(1H)	32
ATU4E	Cycle Register E11	CYLRE11	FFFF $FF00_H$	32	FFE6 2248_H	1(1H)	32
ATU4E	Duty Register E11	DTRE11	FFFF $FF00_H$	32	FFE6 $224C_H$	1(1H)	32
ATU4E	Cycle Reload Register E11	CRLDE11	FFFF $FF00_H$	32	FFE6 2250_H	1(1H)	32
ATU4E	Duty Reload Register E11	DRLDE11	FFFF $FF00_H$	32	FFE6 2254_H	1(1H)	32
ATU4E	Timer counter E12	TCNTE12	0000 0100_H	32	FFE6 2264_H	1(1H)	32
ATU4E	Cycle Register E12	CYLRE12	FFFF $FF00_H$	32	FFE6 2268_H	1(1H)	32
ATU4E	Duty Register E12	DTRE12	FFFF $FF00_H$	32	FFE6 $226C_H$	1(1H)	32

(86/201)

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
ATU4E	Cycle Reload Register E12	CRLDE12	FFFF $FF00_H$	32	FFE6 2270_H	1(1H)	32
ATU4E	Duty Reload Register E12	DRLDE12	FFFF $FF00_H$	32	FFE6 2274_H	1(1H)	32
ATU4E	Timer counter E13	TCNTE13	0000 0100_H	32	FFE6 2284_H	1(1H)	32
ATU4E	Cycle Register E13	CYLRE13	FFFF $FF00_H$	32	FFE6 2288_H	1(1H)	32
ATU4E	Duty Register E13	DTRE13	FFFF $FF00_H$	32	FFE6 $228C_H$	1(1H)	32
ATU4E	Cycle Reload Register E13	CRLDE13	FFFF $FF00_H$	32	FFE6 2290_H	1(1H)	32
ATU4E	Duty Reload Register E13	DRLDE13	FFFF $FF00_H$	32	FFE6 2294_H	1(1H)	32
ATU4E	Sub-block Start Register E2	SSTRE2	00_H	8	FFE6 2300_H	1(1H)	8
ATU4E	Prescaler Register E2	PSCRE2	00_H	8	FFE6 2304_H	1(1H)	8
ATU4E	Timer Control Register E2	TCRE2	00_H	8	FFE6 2308_H	1(1H)	8
ATU4E	Reload Control Register E2	RLDCRE2	00_H	8	FFE6 $230A_H$	1(1H)	8
ATU4E	Output Shutoff Control Register E2	POECRE2	0000_H	16	FFE6 $230C_H$	1(1H)	16
ATU4E	Output Shutoff Level Setting Register E2	SOLVLE2	00_H	8	FFE6 $230E_H$	1(1H)	8
ATU4E	Timer Status Register E2	TSRE2	0000_H	16	FFE6 2310_H	1(1H)	8, 16
ATU4E	Timer Status Clear Register E2	TSCRE2	0000_H	16	FFE6 2312_H	1(1H)	8, 16
ATU4E	Timer Output Control Register E2	TOCRE2	00_H	8	FFE6 2314_H	1(1H)	8
ATU4E	Timer Interrupt Enable Register E2	TIERE2	0000_H	16	FFE6 2316_H	1(1H)	8, 16
ATU4E	Prescaler Channel Register E20	PSCCRE20	00_H	8	FFE6 2318_H	1(1H)	8
ATU4E	Prescaler Channel Register E21	PSCCRE21	00_H	8	FFE6 2319_H	1(1H)	8
ATU4E	Prescaler Channel Register E22	PSCCRE22	00_H	8	FFE6 $231A_H$	1(1H)	8
ATU4E	Prescaler Channel Register E23	PSCCRE23	00_H	8	FFE6 $231B_H$	1(1H)	8
ATU4E	Timer counter E20	TCNTE20	0000 0100_H	32	FFE6 2324_H	1(1H)	32
ATU4E	Cycle Register E20	CYLRE20	FFFF $FF00_H$	32	FFE6 2328_H	1(1H)	32
ATU4E	Duty Register E20	DTRE20	FFFF $FF00_H$	32	FFE6 $232C_H$	1(1H)	32
ATU4E	Cycle Reload Register E20	CRLDE20	FFFF $FF00_H$	32	FFE6 2330_H	1(1H)	32
ATU4E	Duty Reload Register E20	DRLDE20	FFFF $FF00_H$	32	FFE6 2334_H	1(1H)	32
ATU4E	Timer counter E21	TCNTE21	0000 0100_H	32	FFE6 2344_H	1(1H)	32
ATU4E	Cycle Register E21	CYLRE21	FFFF $FF00_H$	32	FFE6 2348_H	1(1H)	32
ATU4E	Duty Register E21	DTRE21	FFFF $FF00_H$	32	FFE6 $234C_H$	1(1H)	32
ATU4E	Cycle Reload Register E21	CRLDE21	FFFF $FF00_H$	32	FFE6 2350_H	1(1H)	32
ATU4E	Duty Reload Register E21	DRLDE21	FFFF $FF00_H$	32	FFE6 2354_H	1(1H)	32
ATU4E	Timer counter E22	TCNTE22	0000 0100_H	32	FFE6 2364_H	1(1H)	32
ATU4E	Cycle Register E22	CYLRE22	FFFF $FF00_H$	32	FFE6 2368_H	1(1H)	32
ATU4E	Duty Register E22	DTRE22	FFFF $FF00_H$	32	FFE6 $236C_H$	1(1H)	32
ATU4E	Cycle Reload Register E22	CRLDE22	FFFF $FF00_H$	32	FFE6 2370_H	1(1H)	32
ATU4E	Duty Reload Register E22	DRLDE22	FFFF $FF00_H$	32	FFE6 2374_H	1(1H)	32
ATU4E	Timer counter E23	TCNTE23	0000 0100_H	32	FFE6 2384_H	1(1H)	32
ATU4E	Cycle Register E23	CYLRE23	FFFF $FF00_H$	32	FFE6 2388_H	1(1H)	32
ATU4E	Duty Register E23	DTRE23	FFFF $FF00_H$	32	FFE6 $238C_H$	1(1H)	32
ATU4E	Cycle Reload Register E23	CRLDE23	FFFF $FF00_H$	32	FFE6 2390_H	1(1H)	32
ATU4E	Duty Reload Register E23	DRLDE23	FFFF $FF00_H$	32	FFE6 2394_H	1(1H)	32
ATU4E	Sub-block Start Register E3	SSTRE3	00_H	8	FFE6 2400_H	1(1H)	8
ATU4E	Prescaler Register E3	PSCRE3	00_H	8	FFE6 2404_H	1(1H)	8
ATU4E	Timer Control Register E3	TCRE3	00_H	8	FFE6 2408_H	1(1H)	8
ATU4E	Reload Control Register E3	RLDCRE3	00_H	8	FFE6 $240A_H$	1(1H)	8
ATU4E	Output Shutoff Control Register E3	POECRE3	0000_H	16	FFE6 $240C_H$	1(1H)	16
ATU4E	Output Shutoff Level Setting Register E3	SOLVLE3	00_H	8	FFE6 $240E_H$	1(1H)	8

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
ATU4E	Timer Status Register E3	TSRE3	0000_H	16	FFE6 2410_H	1(1H)	8, 16
ATU4E	Timer Status Clear Register E3	TSCRE3	0000_H	16	FFE6 2412_H	1(1H)	8, 16
ATU4E	Timer Output Control Register E3	TOCRE3	00_H	8	FFE6 2414_H	1(1H)	8
ATU4E	Timer Interrupt Enable Register E3	TIERE3	0000_H	16	FFE6 2416_H	1(1H)	8, 16
ATU4E	Prescaler Channel Register E30	PSCCRE30	00_H	8	FFE6 2418_H	1(1H)	8
ATU4E	Prescaler Channel Register E31	PSCCRE31	00_H	8	FFE6 2419_H	1(1H)	8
ATU4E	Prescaler Channel Register E32	PSCCRE32	00_H	8	FFE6 $241A_H$	1(1H)	8
ATU4E	Prescaler Channel Register E33	PSCCRE33	00_H	8	FFE6 $241B_H$	1(1H)	8
ATU4E	Timer counter E30	TCNTE30	0000 0100_H	32	FFE6 2424_H	1(1H)	32
ATU4E	Cycle Register E30	CYLRE30	FFFF $FF00_H$	32	FFE6 2428_H	1(1H)	32
ATU4E	Duty Register E30	DTRE30	FFFF $FF00_H$	32	FFE6 $242C_H$	1(1H)	32
ATU4E	Cycle Reload Register E30	CRLDE30	FFFF $FF00_H$	32	FFE6 2430_H	1(1H)	32
ATU4E	Duty Reload Register E30	DRLDE30	FFFF $FF00_H$	32	FFE6 2434_H	1(1H)	32
ATU4E	Timer counter E31	TCNTE31	0000 0100_H	32	FFE6 2444_H	1(1H)	32
ATU4E	Cycle Register E31	CYLRE31	FFFF $FF00_H$	32	FFE6 2448_H	1(1H)	32
ATU4E	Duty Register E31	DTRE31	FFFF $FF00_H$	32	FFE6 $244C_H$	1(1H)	32
ATU4E	Cycle Reload Register E31	CRLDE31	FFFF $FF00_H$	32	FFE6 2450_H	1(1H)	32
ATU4E	Duty Reload Register E31	DRLDE31	FFFF $FF00_H$	32	FFE6 2454_H	1(1H)	32
ATU4E	Timer counter E32	TCNTE32	0000 0100_H	32	FFE6 2464_H	1(1H)	32
ATU4E	Cycle Register E32	CYLRE32	FFFF $FF00_H$	32	FFE6 2468_H	1(1H)	32
ATU4E	Duty Register E32	DTRE32	FFFF $FF00_H$	32	FFE6 $246C_H$	1(1H)	32
ATU4E	Cycle Reload Register E32	CRLDE32	FFFF $FF00_H$	32	FFE6 2470_H	1(1H)	32
ATU4E	Duty Reload Register E32	DRLDE32	FFFF $FF00_H$	32	FFE6 2474_H	1(1H)	32
ATU4E	Timer counter E33	TCNTE33	0000 0100_H	32	FFE6 2484_H	1(1H)	32
ATU4E	Cycle Register E33	CYLRE33	FFFF $FF00_H$	32	FFE6 2488_H	1(1H)	32
ATU4E	Duty Register E33	DTRE33	FFFF $FF00_H$	32	FFE6 $248C_H$	1(1H)	32
ATU4E	Cycle Reload Register E33	CRLDE33	FFFF $FF00_H$	32	FFE6 2490_H	1(1H)	32
ATU4E	Duty Reload Register E33	DRLDE33	FFFF $FF00_H$	32	FFE6 2494_H	1(1H)	32
ATU4E	Sub-block Start Register E4	SSTRE4	00_H	8	FFE6 2500_H	1(1H)	8
ATU4E	Prescaler Register E4	PSCRE4	00_H	8	FFE6 2504_H	1(1H)	8
ATU4E	Timer Control Register E4	TCRE4	00_H	8	FFE6 2508_H	1(1H)	8
ATU4E	Reload Control Register E4	RLDCRE4	00_H	8	FFE6 $250A_H$	1(1H)	8
ATU4E	Output Shutoff Control Register E4	POECRE4	0000_H	16	FFE6 $250C_H$	1(1H)	16
ATU4E	Output Shutoff Level Setting Register E4	SOLVLE4	00_H	8	FFE6 $250E_H$	1(1H)	8
ATU4E	Timer Status Register E4	TSRE4	0000_H	16	FFE6 2510_H	1(1H)	8, 16
ATU4E	Timer Status Clear Register E4	TSCRE4	0000_H	16	FFE6 2512_H	1(1H)	8, 16
ATU4E	Timer Output Control Register E4	TOCRE4	00_H	8	FFE6 2514_H	1(1H)	8
ATU4E	Timer Interrupt Enable Register E4	TIERE4	0000_H	16	FFE6 2516_H	1(1H)	8, 16
ATU4E	Prescaler Channel Register E40	PSCCRE40	00_H	8	FFE6 2518_H	1(1H)	8
ATU4E	Prescaler Channel Register E41	PSCCRE41	00_H	8	FFE6 2519_H	1(1H)	8
ATU4E	Prescaler Channel Register E42	PSCCRE42	00_H	8	FFE6 $251A_H$	1(1H)	8
ATU4E	Prescaler Channel Register E43	PSCCRE43	00_H	8	FFE6 $251B_H$	1(1H)	8
ATU4E	Timer counter E40	TCNTE40	0000 0100_H	32	FFE6 2524_H	1(1H)	32
ATU4E	Cycle Register E40	CYLRE40	FFFF $FF00_H$	32	FFE6 2528_H	1(1H)	32
ATU4E	Duty Register E40	DTRE40	FFFF $FF00_H$	32	FFE6 $252C_H$	1(1H)	32
ATU4E	Cycle Reload Register E40	CRLDE40	FFFF $FF00_H$	32	FFE6 2530_H	1(1H)	32
ATU4E	Duty Reload Register E40	DRLDE40	FFFF $FF00_H$	32	FFE6 2534_H	1(1H)	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
ATU4E	Timer counter E41	TCNTE41	0000 0100_H	32	FFE6 2544_H	1(1H)	32
ATU4E	Cycle Register E41	CYLRE41	FFFF $FF00_H$	32	FFE6 2548_H	1(1H)	32
ATU4E	Duty Register E41	DTRE41	FFFF $FF00_H$	32	FFE6 $254C_H$	1(1H)	32
ATU4E	Cycle Reload Register E41	CRLDE41	FFFF $FF00_H$	32	FFE6 2550_H	1(1H)	32
ATU4E	Duty Reload Register E41	DRLDE41	FFFF $FF00_H$	32	FFE6 2554_H	1(1H)	32
ATU4E	Timer counter E42	TCNTE42	0000 0100_H	32	FFE6 2564_H	1(1H)	32
ATU4E	Cycle Register E42	CYLRE42	FFFF $FF00_H$	32	FFE6 2568_H	1(1H)	32
ATU4E	Duty Register E42	DTRE42	FFFF $FF00_H$	32	FFE6 $256C_H$	1(1H)	32
ATU4E	Cycle Reload Register E42	CRLDE42	FFFF $FF00_H$	32	FFE6 2570_H	1(1H)	32
ATU4E	Duty Reload Register E42	DRLDE42	FFFF $FF00_H$	32	FFE6 2574_H	1(1H)	32
ATU4E	Timer counter E43	TCNTE43	0000 0100_H	32	FFE6 2584_H	1(1H)	32
ATU4E	Cycle Register E43	CYLRE43	FFFF $FF00_H$	32	FFE6 2588_H	1(1H)	32
ATU4E	Duty Register E43	DTRE43	FFFF $FF00_H$	32	FFE6 $258C_H$	1(1H)	32
ATU4E	Cycle Reload Register E43	CRLDE43	FFFF $FF00_H$	32	FFE6 2590_H	1(1H)	32
ATU4E	Duty Reload Register E43	DRLDE43	FFFF $FF00_H$	32	FFE6 2594_H	1(1H)	32
ATU4E	Sub-block Start Register E5	SSTRE5	00_H	8	FFE6 2600_H	1(1H)	8
ATU4E	Prescaler Register E5	PSCRE5	00_H	8	FFE6 2604_H	1(1H)	8
ATU4E	Timer Control Register E5	TCRE5	00_H	8	FFE6 2608_H	1(1H)	8
ATU4E	Reload Control Register E5	RLDCRE5	00_H	8	FFE6 $260A_H$	1(1H)	8
ATU4E	Output Shutoff Control Register E5	POECRE5	0000_H	16	FFE6 $260C_H$	1(1H)	16
ATU4E	Output Shutoff Level Setting Register E5	SOLVLE5	00_H	8	FFE6 $260E_H$	1(1H)	8
ATU4E	Timer Status Register E5	TSRE5	0000_H	16	FFE6 2610_H	1(1H)	8, 16
ATU4E	Timer Status Clear Register E5	TSCRE5	0000_H	16	FFE6 2612_H	1(1H)	8, 16
ATU4E	Timer Output Control Register E5	TOCRE5	00_H	8	FFE6 2614_H	1(1H)	8
ATU4E	Timer Interrupt Enable Register E5	TIERE5	0000_H	16	FFE6 2616_H	1(1H)	8, 16
ATU4E	Prescaler Channel Register E50	PSCCRE50	00_H	8	FFE6 2618_H	1(1H)	8
ATU4E	Prescaler Channel Register E51	PSCCRE51	00_H	8	FFE6 2619_H	1(1H)	8
ATU4E	Prescaler Channel Register E52	PSCCRE52	00_H	8	FFE6 $261A_H$	1(1H)	8
ATU4E	Prescaler Channel Register E53	PSCCRE53	00_H	8	FFE6 $261B_H$	1(1H)	8
ATU4E	Timer counter E50	TCNTE50	0000 0100_H	32	FFE6 2624_H	1(1H)	32
ATU4E	Cycle Register E50	CYLRE50	FFFF $FF00_H$	32	FFE6 2628_H	1(1H)	32
ATU4E	Duty Register E50	DTRE50	FFFF $FF00_H$	32	FFE6 $262C_H$	1(1H)	32
ATU4E	Cycle Reload Register E50	CRLDE50	FFFF $FF00_H$	32	FFE6 2630_H	1(1H)	32
ATU4E	Duty Reload Register E50	DRLDE50	FFFF $FF00_H$	32	FFE6 2634_H	1(1H)	32
ATU4E	Timer counter E51	TCNTE51	0000 0100_H	32	FFE6 2644_H	1(1H)	32
ATU4E	Cycle Register E51	CYLRE51	FFFF $FF00_H$	32	FFE6 2648_H	1(1H)	32
ATU4E	Duty Register E51	DTRE51	FFFF $FF00_H$	32	FFE6 $264C_H$	1(1H)	32
ATU4E	Cycle Reload Register E51	CRLDE51	FFFF $FF00_H$	32	FFE6 2650_H	1(1H)	32
ATU4E	Duty Reload Register E51	DRLDE51	FFFF $FF00_H$	32	FFE6 2654_H	1(1H)	32
ATU4E	Timer counter E52	TCNTE52	0000 0100_H	32	FFE6 2664_H	1(1H)	32
ATU4E	Cycle Register E52	CYLRE52	FFFF $FF00_H$	32	FFE6 2668_H	1(1H)	32
ATU4E	Duty Register E52	DTRE52	FFFF $FF00_H$	32	FFE6 $266C_H$	1(1H)	32
ATU4E	Cycle Reload Register E52	CRLDE52	FFFF $FF00_H$	32	FFE6 2670_H	1(1H)	32
ATU4E	Duty Reload Register E52	DRLDE52	FFFF $FF00_H$	32	FFE6 2674_H	1(1H)	32
ATU4E	Timer counter E53	TCNTE53	0000 0100_H	32	FFE6 2684_H	1(1H)	32
ATU4E	Cycle Register E53	CYLRE53	FFFF $FF00_H$	32	FFE6 2688_H	1(1H)	32
ATU4E	Duty Register E53	DTRE53	FFFF $FF00_H$	32	FFE6 $268C_H$	1(1H)	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
ATU4E	Cycle Reload Register E53	CRLDE53	FFFF $FF00_H$	32	FFE6 2690_H	1(1H)	32
ATU4E	Duty Reload Register E53	DRLDE53	FFFF $FF00_H$	32	FFE6 2694_H	1(1H)	32
ATU4E	Sub-block Start Register E6	SSTRE6	00_H	8	FFE6 2700_H	1(1H)	8
ATU4E	Prescaler Register E6	PSCRE6	00_H	8	FFE6 2704_H	1(1H)	8
ATU4E	Timer Control Register E6	TCRE6	00_H	8	FFE6 2708_H	1(1H)	8
ATU4E	Reload Control Register E6	RLDCRE6	00_H	8	FFE6 $270A_H$	1(1H)	8
ATU4E	Output Shutoff Control Register E6	POECRE6	0000_H	16	FFE6 $270C_H$	1(1H)	16
ATU4E	Output Shutoff Level Setting Register E6	SOLVLE6	00_H	8	FFE6 $270E_H$	1(1H)	8
ATU4E	Timer Status Register E6	TSRE6	0000_H	16	FFE6 2710_H	1(1H)	8, 16
ATU4E	Timer Status Clear Register E6	TSCRE6	0000_H	16	FFE6 2712_H	1(1H)	8, 16
ATU4E	Timer Output Control Register E6	TOCRE6	00_H	8	FFE6 2714_H	1(1H)	8
ATU4E	Timer Interrupt Enable Register E6	TIERE6	0000_H	16	FFE6 2716_H	1(1H)	8, 16
ATU4E	Prescaler Channel Register E60	PSCCRE60	00_H	8	FFE6 2718_H	1(1H)	8
ATU4E	Prescaler Channel Register E61	PSCCRE61	00_H	8	FFE6 2719_H	1(1H)	8
ATU4E	Prescaler Channel Register E62	PSCCRE62	00_H	8	FFE6 $271A_H$	1(1H)	8
ATU4E	Prescaler Channel Register E63	PSCCRE63	00_H	8	FFE6 $271B_H$	1(1H)	8
ATU4E	Timer counter E60	TCNTE60	0000 0100_H	32	FFE6 2724_H	1(1H)	32
ATU4E	Cycle Register E60	CYLRE60	FFFF $FF00_H$	32	FFE6 2728_H	1(1H)	32
ATU4E	Duty Register E60	DTRE60	FFFF $FF00_H$	32	FFE6 $272C_H$	1(1H)	32
ATU4E	Cycle Reload Register E60	CRLDE60	FFFF $FF00_H$	32	FFE6 2730_H	1(1H)	32
ATU4E	Duty Reload Register E60	DRLDE60	FFFF $FF00_H$	32	FFE6 2734_H	1(1H)	32
ATU4E	Timer counter E61	TCNTE61	0000 0100_H	32	FFE6 2744_H	1(1H)	32
ATU4E	Cycle Register E61	CYLRE61	FFFF $FF00_H$	32	FFE6 2748_H	1(1H)	32
ATU4E	Duty Register E61	DTRE61	FFFF $FF00_H$	32	FFE6 $274C_H$	1(1H)	32
ATU4E	Cycle Reload Register E61	CRLDE61	FFFF $FF00_H$	32	FFE6 2750_H	1(1H)	32
ATU4E	Duty Reload Register E61	DRLDE61	FFFF $FF00_H$	32	FFE6 2754_H	1(1H)	32
ATU4E	Timer counter E62	TCNTE62	0000 0100_H	32	FFE6 2764_H	1(1H)	32
ATU4E	Cycle Register E62	CYLRE62	FFFF $FF00_H$	32	FFE6 2768_H	1(1H)	32
ATU4E	Duty Register E62	DTRE62	FFFF $FF00_H$	32	FFE6 $276C_H$	1(1H)	32
ATU4E	Cycle Reload Register E62	CRLDE62	FFFF $FF00_H$	32	FFE6 2770_H	1(1H)	32
ATU4E	Duty Reload Register E62	DRLDE62	FFFF $FF00_H$	32	FFE6 2774_H	1(1H)	32
ATU4E	Timer counter E63	TCNTE63	0000 0100_H	32	FFE6 2784_H	1(1H)	32
ATU4E	Cycle Register E63	CYLRE63	FFFF $FF00_H$	32	FFE6 2788_H	1(1H)	32
ATU4E	Duty Register E63	DTRE63	FFFF $FF00_H$	32	FFE6 $278C_H$	1(1H)	32
ATU4E	Cycle Reload Register E63	CRLDE63	FFFF $FF00_H$	32	FFE6 2790_H	1(1H)	32
ATU4E	Duty Reload Register E63	DRLDE63	FFFF $FF00_H$	32	FFE6 2794_H	1(1H)	32
ATU4F	Timer Start Register F	TSTRF	0000_H	16	FFE6 3000_H	1(1H)	8, 16
ATU4F	Noise Cancel Mode Channel Register 1F	NCMCR1F	0000_H	16	FFE6 3004_H	1(1H)	8, 16
ATU4F	Noise Cancel Mode Channel Register 2F	NCMCR2F	0000_H	16	FFE6 3008_H	1(1H)	8, 16
ATU4F	Noise Canceller Control Register F	NCCRF	0000_H	16	FFE6 $300C_H$	1(1H)	8, 16
ATU4F	Private Function Control Register F	PVFCRF	0000_H	16	FFE6 3010_H	1(1H)	16
ATU4F	Timer Control Register 1F0	TCR1F0	00_H	8	FFE6 3040_H	1(1H)	8
ATU4F	Timer Control Register 2F0	TCR2F0	00_H	8	FFE6 3042_H	1(1H)	8
ATU4F	Back-up Control Register F0	BKCRF0	00_H	8	FFE6 3044_H	1(1H)	8
ATU4F	Timer Status Register F0	TSRF0	00_H	8	FFE6 3045_H	1(1H)	8
ATU4F	Timer Status Clear Register F0	TSCRF0	00_H	8	FFE6 3046_H	1(1H)	8

(90/201)

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
ATU4F	Timer Interrupt Enable Register F0	TIERF0	00_H	8	FFE6 3047_H	1(1H)	8
ATU4F	Noise Cancel Counter FA00	NCNTFA0	0000_H	16	FFE6 3048_H	1(1H)	16
ATU4F	Noise cancel Register FA00	NCRFA0	0000_H	16	FFE6 $304A_H$	1(1H)	16
ATU4F	Noise Cancel Counter FB00	NCNTFB0	0000_H	16	FFE6 $304C_H$	1(1H)	16
ATU4F	Noise cancel Register FB00	NCRFB0	0000_H	16	FFE6 $304E_H$	1(1H)	16
ATU4F	Timer Counter AF0	ECNTAF0	$0000\ 0000_H$	32	FFE6 3050_H	1(1H)	32
ATU4F	General Register AF0	GRAF0	$FFFF\ FFFF_H$	32	FFE6 3054_H	1(1H)	32
ATU4F	Event Counter F0	ECNTBF0	0000_H	16	FFE6 3058_H	1(1H)	16
ATU4F	General Register BF0	GRBF0	$FFFF_H$	16	FFE6 $305C_H$	1(1H)	16
ATU4F	Timer Counter CF0	ECNTCF0	$0000\ 0000_H$	32	FFE6 3060_H	1(1H)	32
ATU4F	Back-up Register CF0	BGRCF0	$FFFF\ FFFF_H$	32	FFE6 3064_H	1(1H)	32
ATU4F	General Register CF0	GRCF0	$FFFF\ FFFF_H$	32	FFE6 3064_H	1(1H)	32
ATU4F	General Register DF0	GRDF0	$FFFF\ FFFF_H$	32	FFE6 3068_H	1(1H)	32
ATU4F	Capture Output Register F0	CDRF0	$0000\ FFFF_H$	32	FFE6 $306C_H$	1(1H)	32
ATU4F	Timer Control Register 1F1	TCR1F1	00_H	8	FFE6 3080_H	1(1H)	8
ATU4F	Timer Control Register 2F1	TCR2F1	00_H	8	FFE6 3082_H	1(1H)	8
ATU4F	Back-up Control Register F1	BKCRF1	00_H	8	FFE6 3084_H	1(1H)	8
ATU4F	Timer Status Register F1	TSRF1	00_H	8	FFE6 3085_H	1(1H)	8
ATU4F	Timer Status Clear Register F1	TSCRF1	00_H	8	FFE6 3086_H	1(1H)	8
ATU4F	Timer Interrupt Enable Register F1	TIERF1	00_H	8	FFE6 3087_H	1(1H)	8
ATU4F	Noise Cancel Counter FA01	NCNTFA1	0000_H	16	FFE6 3088_H	1(1H)	16
ATU4F	Noise cancel Register FA01	NCRFA1	0000_H	16	FFE6 $308A_H$	1(1H)	16
ATU4F	Noise Cancel Counter FB01	NCNTFB1	0000_H	16	FFE6 $308C_H$	1(1H)	16
ATU4F	Noise cancel Register FB01	NCRFB1	0000_H	16	FFE6 $308E_H$	1(1H)	16
ATU4F	Timer Counter AF1	ECNTAF1	$0000\ 0000_H$	32	FFE6 3090_H	1(1H)	32
ATU4F	General Register AF1	GRAF1	$FFFF\ FFFF_H$	32	FFE6 3094_H	1(1H)	32
ATU4F	Event Counter F1	ECNTBF1	0000_H	16	FFE6 3098_H	1(1H)	16
ATU4F	General Register BF1	GRBF1	$FFFF_H$	16	FFE6 $309C_H$	1(1H)	16
ATU4F	Timer Counter CF1	ECNTCF1	$0000\ 0000_H$	32	FFE6 $30A0_H$	1(1H)	32
ATU4F	Back-up Register CF1	BGRCF1	$FFFF\ FFFF_H$	32	FFE6 $30A4_H$	1(1H)	32
ATU4F	General Register CF1	GRCF1	$FFFF\ FFFF_H$	32	FFE6 $30A4_H$	1(1H)	32
ATU4F	General Register DF1	GRDF1	$FFFF\ FFFF_H$	32	FFE6 $30A8_H$	1(1H)	32
ATU4F	Capture Output Register F1	CDRF1	$0000\ FFFF_H$	32	FFE6 $30AC_H$	1(1H)	32
ATU4F	Timer Control Register 1F2	TCR1F2	00_H	8	FFE6 $30C0_H$	1(1H)	8
ATU4F	Timer Control Register 2F2	TCR2F2	00_H	8	FFE6 $30C2_H$	1(1H)	8
ATU4F	Back-up Control Register F2	BKCRF2	00_H	8	FFE6 $30C4_H$	1(1H)	8
ATU4F	Timer Status Register F2	TSRF2	00_H	8	FFE6 $30C5_H$	1(1H)	8
ATU4F	Timer Status Clear Register F2	TSCRF2	00_H	8	FFE6 $30C6_H$	1(1H)	8
ATU4F	Timer Interrupt Enable Register F2	TIERF2	00_H	8	FFE6 $30C7_H$	1(1H)	8
ATU4F	Noise Cancel Counter FA02	NCNTFA2	0000_H	16	FFE6 $30C8_H$	1(1H)	16
ATU4F	Noise cancel Register FA02	NCRFA2	0000_H	16	FFE6 $30CA_H$	1(1H)	16
ATU4F	Noise Cancel Counter FB02	NCNTFB2	0000_H	16	FFE6 $30CC_H$	1(1H)	16
ATU4F	Noise cancel Register FB02	NCRFB2	0000_H	16	FFE6 $30CE_H$	1(1H)	16
ATU4F	Timer Counter AF2	ECNTAF2	$0000\ 0000_H$	32	FFE6 $30D0_H$	1(1H)	32
ATU4F	General Register AF2	GRAF2	$FFFF\ FFFF_H$	32	FFE6 $30D4_H$	1(1H)	32
ATU4F	Event Counter F2	ECNTBF2	0000_H	16	FFE6 $30D8_H$	1(1H)	16
ATU4F	General Register BF2	GRBF2	$FFFF_H$	16	FFE6 $30DC_H$	1(1H)	16

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
ATU4F	Timer Counter CF2	ECNTCF2	0000 0000_H	32	FFE6 $30E0_H$	1(1H)	32
ATU4F	Back-up Register CF2	BGRCF2	FFFF $FFFF_H$	32	FFE6 $30E4_H$	1(1H)	32
ATU4F	General Register CF2	GRCF2	FFFF $FFFF_H$	32	FFE6 $30E4_H$	1(1H)	32
ATU4F	General Register DF2	GRDF2	FFFF $FFFF_H$	32	FFE6 $30E8_H$	1(1H)	32
ATU4F	Capture Output Register F2	CDRF2	0000 $FFFF_H$	32	FFE6 $30EC_H$	1(1H)	32
ATU4F	Timer Control Register 1F3	TCR1F3	00_H	8	FFE6 3100_H	1(1H)	8
ATU4F	Timer Control Register 2F3	TCR2F3	00_H	8	FFE6 3102_H	1(1H)	8
ATU4F	Back-up Control Register F3	BKCRF3	00_H	8	FFE6 3104_H	1(1H)	8
ATU4F	Timer Status Register F3	TSRF3	00_H	8	FFE6 3105_H	1(1H)	8
ATU4F	Timer Status Clear Register F3	TSCRF3	00_H	8	FFE6 3106_H	1(1H)	8
ATU4F	Timer Interrupt Enable Register F3	TIERF3	00_H	8	FFE6 3107_H	1(1H)	8
ATU4F	Noise Cancel Counter FA03	NCNTFA3	0000_H	16	FFE6 3108_H	1(1H)	16
ATU4F	Noise cancel Register FA03	NCRFA3	0000_H	16	FFE6 $310A_H$	1(1H)	16
ATU4F	Timer Counter AF3	ECNTAF3	0000 0000_H	32	FFE6 3110_H	1(1H)	32
ATU4F	Back-up Register AF3	BGRAF3	FFFF $FFFF_H$	32	FFE6 3114_H	1(1H)	32
ATU4F	General Register AF3	GRAF3	FFFF $FFFF_H$	32	FFE6 3114_H	1(1H)	32
ATU4F	Event Counter F3	ECNTBF3	0000_H	16	FFE6 3118_H	1(1H)	16
ATU4F	General Register BF3	GRBF3	$FFFF_H$	16	FFE6 $311C_H$	1(1H)	16
ATU4F	Timer Counter CF3	ECNTCF3	0000 0000_H	32	FFE6 3120_H	1(1H)	32
ATU4F	Back-up Register CF3	BGRCF3	FFFF $FFFF_H$	32	FFE6 3124_H	1(1H)	32
ATU4F	General Register CF3	GRCF3	FFFF $FFFF_H$	32	FFE6 3124_H	1(1H)	32
ATU4F	General Register DF3	GRDF3	FFFF $FFFF_H$	32	FFE6 3128_H	1(1H)	32
ATU4F	Back-up Register DF3	BGRDF3	FFFF $FFFF_H$	32	FFE6 3128_H	1(1H)	32
ATU4F	Capture Output Register F3	CDRF3	0000 $FFFF_H$	32	FFE6 $312C_H$	1(1H)	32
ATU4F	Timer Control Register 1F4	TCR1F4	00_H	8	FFE6 3140_H	1(1H)	8
ATU4F	Timer Control Register 2F4	TCR2F4	00_H	8	FFE6 3142_H	1(1H)	8
ATU4F	Back-up Control Register F4	BKCRF4	00_H	8	FFE6 3144_H	1(1H)	8
ATU4F	Timer Status Register F4	TSRF4	00_H	8	FFE6 3145_H	1(1H)	8
ATU4F	Timer Status Clear Register F4	TSCRF4	00_H	8	FFE6 3146_H	1(1H)	8
ATU4F	Timer Interrupt Enable Register F4	TIERF4	00_H	8	FFE6 3147_H	1(1H)	8
ATU4F	Noise Cancel Counter FA04	NCNTFA4	0000_H	16	FFE6 3148_H	1(1H)	16
ATU4F	Noise cancel Register FA04	NCRFA4	0000_H	16	FFE6 $314A_H$	1(1H)	16
ATU4F	Timer Counter AF4	ECNTAF4	0000 0000_H	32	FFE6 3150_H	1(1H)	32
ATU4F	Back-up Register AF4	BGRAF4	FFFF $FFFF_H$	32	FFE6 3154_H	1(1H)	32
ATU4F	General Register AF4	GRAF4	FFFF $FFFF_H$	32	FFE6 3154_H	1(1H)	32
ATU4F	Event Counter F4	ECNTBF4	0000_H	16	FFE6 3158_H	1(1H)	16
ATU4F	General Register BF4	GRBF4	$FFFF_H$	16	FFE6 $315C_H$	1(1H)	16
ATU4F	Timer Counter CF4	ECNTCF4	0000 0000_H	32	FFE6 3160_H	1(1H)	32
ATU4F	Back-up Register CF4	BGRCF4	FFFF $FFFF_H$	32	FFE6 3164_H	1(1H)	32
ATU4F	General Register CF4	GRCF4	FFFF $FFFF_H$	32	FFE6 3164_H	1(1H)	32
ATU4F	General Register DF4	GRDF4	FFFF $FFFF_H$	32	FFE6 3168_H	1(1H)	32
ATU4F	Back-up Register DF4	BGRDF4	FFFF $FFFF_H$	32	FFE6 3168_H	1(1H)	32
ATU4F	Capture Output Register F4	CDRF4	0000 $FFFF_H$	32	FFE6 $316C_H$	1(1H)	32
ATU4F	Timer Control Register 1F5	TCR1F5	00_H	8	FFE6 3180_H	1(1H)	8
ATU4F	Timer Control Register 2F5	TCR2F5	00_H	8	FFE6 3182_H	1(1H)	8
ATU4F	Back-up Control Register F5	BKCRF5	00_H	8	FFE6 3184_H	1(1H)	8
ATU4F	Timer Status Register F5	TSRF5	00_H	8	FFE6 3185_H	1(1H)	8

(92/201)

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
ATU4F	Timer Status Clear Register F5	TSCRF5	00_H	8	FFE6 3186$_H$	1(1H)	8
ATU4F	Timer Interrupt Enable Register F5	TIERF5	00_H	8	FFE6 3187$_H$	1(1H)	8
ATU4F	Noise Cancel Counter FA05	NCNTFA5	0000_H	16	FFE6 3188$_H$	1(1H)	16
ATU4F	Noise cancel Register FA05	NCRFA5	0000_H	16	FFE6 318A$_H$	1(1H)	16
ATU4F	Timer Counter AF5	ECNTAF5	0000 0000_H	32	FFE6 3190$_H$	1(1H)	32
ATU4F	Back-up Register AF5	BGRAF5	FFFF $FFFF_H$	32	FFE6 3194$_H$	1(1H)	32
ATU4F	General Register AF5	GRAF5	FFFF $FFFF_H$	32	FFE6 3194$_H$	1(1H)	32
ATU4F	Event Counter F5	ECNTBF5	0000_H	16	FFE6 3198$_H$	1(1H)	16
ATU4F	General Register BF5	GRBF5	$FFFF_H$	16	FFE6 319C$_H$	1(1H)	16
ATU4F	Timer Counter CF5	ECNTCF5	0000 0000_H	32	FFE6 31A0$_H$	1(1H)	32
ATU4F	Back-up Register CF5	BGRCF5	FFFF $FFFF_H$	32	FFE6 31A4$_H$	1(1H)	32
ATU4F	General Register CF5	GRCF5	FFFF $FFFF_H$	32	FFE6 31A4$_H$	1(1H)	32
ATU4F	General Register DF5	GRDF5	FFFF $FFFF_H$	32	FFE6 31A8$_H$	1(1H)	32
ATU4F	Back-up Register DF5	BGRDF5	FFFF $FFFF_H$	32	FFE6 31A8$_H$	1(1H)	32
ATU4F	Capture Output Register F5	CDRF5	0000 $FFFF_H$	32	FFE6 31AC$_H$	1(1H)	32
ATU4F	Timer Control Register 1F6	TCR1F6	00_H	8	FFE6 31C0$_H$	1(1H)	8
ATU4F	Timer Control Register 2F6	TCR2F6	00_H	8	FFE6 31C2$_H$	1(1H)	8
ATU4F	Back-up Control Register F6	BKCRF6	00_H	8	FFE6 31C4$_H$	1(1H)	8
ATU4F	Timer Status Register F6	TSRF6	00_H	8	FFE6 31C5$_H$	1(1H)	8
ATU4F	Timer Status Clear Register F6	TSCRF6	00_H	8	FFE6 31C6$_H$	1(1H)	8
ATU4F	Timer Interrupt Enable Register F6	TIERF6	00_H	8	FFE6 31C7$_H$	1(1H)	8
ATU4F	Noise Cancel Counter FA06	NCNTFA6	0000_H	16	FFE6 31C8$_H$	1(1H)	16
ATU4F	Noise cancel Register FA06	NCRFA6	0000_H	16	FFE6 31CA$_H$	1(1H)	16
ATU4F	Timer Counter AF6	ECNTAF6	0000 0000_H	32	FFE6 31D0$_H$	1(1H)	32
ATU4F	Back-up Register AF6	BGRAF6	FFFF $FFFF_H$	32	FFE6 31D4$_H$	1(1H)	32
ATU4F	General Register AF6	GRAF6	FFFF $FFFF_H$	32	FFE6 31D4$_H$	1(1H)	32
ATU4F	Event Counter F6	ECNTBF6	0000_H	16	FFE6 31D8$_H$	1(1H)	16
ATU4F	General Register BF6	GRBF6	$FFFF_H$	16	FFE6 31DC$_H$	1(1H)	16
ATU4F	Timer Counter CF6	ECNTCF6	0000 0000_H	32	FFE6 31E0$_H$	1(1H)	32
ATU4F	Back-up Register CF6	BGRCF6	FFFF $FFFF_H$	32	FFE6 31E4$_H$	1(1H)	32
ATU4F	General Register CF6	GRCF6	FFFF $FFFF_H$	32	FFE6 31E4$_H$	1(1H)	32
ATU4F	General Register DF6	GRDF6	FFFF $FFFF_H$	32	FFE6 31E8$_H$	1(1H)	32
ATU4F	Back-up Register DF6	BGRDF6	FFFF $FFFF_H$	32	FFE6 31E8$_H$	1(1H)	32
ATU4F	Capture Output Register F6	CDRF6	0000 $FFFF_H$	32	FFE6 31EC$_H$	1(1H)	32
ATU4F	Timer Control Register 1F7	TCR1F7	00_H	8	FFE6 3200$_H$	1(1H)	8
ATU4F	Timer Control Register 2F7	TCR2F7	00_H	8	FFE6 3202$_H$	1(1H)	8
ATU4F	Back-up Control Register F7	BKCRF7	00_H	8	FFE6 3204$_H$	1(1H)	8
ATU4F	Timer Status Register F7	TSRF7	00_H	8	FFE6 3205$_H$	1(1H)	8
ATU4F	Timer Status Clear Register F7	TSCRF7	00_H	8	FFE6 3206$_H$	1(1H)	8
ATU4F	Timer Interrupt Enable Register F7	TIERF7	00_H	8	FFE6 3207$_H$	1(1H)	8
ATU4F	Noise Cancel Counter FA07	NCNTFA7	0000_H	16	FFE6 3208$_H$	1(1H)	16
ATU4F	Noise cancel Register FA07	NCRFA7	0000_H	16	FFE6 320A$_H$	1(1H)	16
ATU4F	Timer Counter AF7	ECNTAF7	0000 0000_H	32	FFE6 3210$_H$	1(1H)	32
ATU4F	Back-up Register AF7	BGRAF7	FFFF $FFFF_H$	32	FFE6 3214$_H$	1(1H)	32
ATU4F	General Register AF7	GRAF7	FFFF $FFFF_H$	32	FFE6 3214$_H$	1(1H)	32
ATU4F	Event Counter F7	ECNTBF7	0000_H	16	FFE6 3218$_H$	1(1H)	16
ATU4F	General Register BF7	GRBF7	$FFFF_H$	16	FFE6 321C$_H$	1(1H)	16

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
ATU4F	Timer Counter CF7	ECNTCF7	0000 0000$_H$	32	FFE6 3220$_H$	1(1H)	32
ATU4F	Back-up Register CF7	BGRCF7	FFFF FFFF$_H$	32	FFE6 3224$_H$	1(1H)	32
ATU4F	General Register CF7	GRCF7	FFFF FFFF$_H$	32	FFE6 3224$_H$	1(1H)	32
ATU4F	General Register DF7	GRDF7	FFFF FFFF$_H$	32	FFE6 3228$_H$	1(1H)	32
ATU4F	Back-up Register DF7	BGRDF7	FFFF FFFF$_H$	32	FFE6 3228$_H$	1(1H)	32
ATU4F	Capture Output Register F7	CDRF7	0000 FFFF$_H$	32	FFE6 322C$_H$	1(1H)	32
ATU4F	Timer Control Register 1F8	TCR1F8	00$_H$	8	FFE6 3240$_H$	1(1H)	8
ATU4F	Timer Control Register 2F8	TCR2F8	00$_H$	8	FFE6 3242$_H$	1(1H)	8
ATU4F	Back-up Control Register F8	BKCRF8	00$_H$	8	FFE6 3244$_H$	1(1H)	8
ATU4F	Timer Status Register F8	TSRF8	00$_H$	8	FFE6 3245$_H$	1(1H)	8
ATU4F	Timer Status Clear Register F8	TSCRF8	00$_H$	8	FFE6 3246$_H$	1(1H)	8
ATU4F	Timer Interrupt Enable Register F8	TIERF8	00$_H$	8	FFE6 3247$_H$	1(1H)	8
ATU4F	Noise Cancel Counter FA08	NCNTFA8	0000$_H$	16	FFE6 3248$_H$	1(1H)	16
ATU4F	Noise cancel Register FA08	NCRFA8	0000$_H$	16	FFE6 324A$_H$	1(1H)	16
ATU4F	Timer Counter AF8	ECNTAF8	0000 0000$_H$	32	FFE6 3250$_H$	1(1H)	32
ATU4F	Back-up Register AF8	BGRAF8	FFFF FFFF$_H$	32	FFE6 3254$_H$	1(1H)	32
ATU4F	General Register AF8	GRAF8	FFFF FFFF$_H$	32	FFE6 3254$_H$	1(1H)	32
ATU4F	Event Counter F8	ECNTBF8	0000$_H$	16	FFE6 3258$_H$	1(1H)	16
ATU4F	General Register BF8	GRBF8	FFFF$_H$	16	FFE6 325C$_H$	1(1H)	16
ATU4F	Timer Counter CF8	ECNTCF8	0000 0000$_H$	32	FFE6 3260$_H$	1(1H)	32
ATU4F	Back-up Register CF8	BGRCF8	FFFF FFFF$_H$	32	FFE6 3264$_H$	1(1H)	32
ATU4F	General Register CF8	GRCF8	FFFF FFFF$_H$	32	FFE6 3264$_H$	1(1H)	32
ATU4F	General Register DF8	GRDF8	FFFF FFFF$_H$	32	FFE6 3268$_H$	1(1H)	32
ATU4F	Back-up Register DF8	BGRDF8	FFFF FFFF$_H$	32	FFE6 3268$_H$	1(1H)	32
ATU4F	Capture Output Register F8	CDRF8	0000 FFFF$_H$	32	FFE6 326C$_H$	1(1H)	32
ATU4F	Timer Control Register 1F9	TCR1F9	00$_H$	8	FFE6 3280$_H$	1(1H)	8
ATU4F	Timer Control Register 2F9	TCR2F9	00$_H$	8	FFE6 3282$_H$	1(1H)	8
ATU4F	Back-up Control Register F9	BKCRF9	00$_H$	8	FFE6 3284$_H$	1(1H)	8
ATU4F	Timer Status Register F9	TSRF9	00$_H$	8	FFE6 3285$_H$	1(1H)	8
ATU4F	Timer Status Clear Register F9	TSCRF9	00$_H$	8	FFE6 3286$_H$	1(1H)	8
ATU4F	Timer Interrupt Enable Register F9	TIERF9	00$_H$	8	FFE6 3287$_H$	1(1H)	8
ATU4F	Noise Cancel Counter FA09	NCNTFA9	0000$_H$	16	FFE6 3288$_H$	1(1H)	16
ATU4F	Noise cancel Register FA09	NCRFA9	0000$_H$	16	FFE6 328A$_H$	1(1H)	16
ATU4F	Timer Counter AF9	ECNTAF9	0000 0000$_H$	32	FFE6 3290$_H$	1(1H)	32
ATU4F	Back-up Register AF9	BGRAF9	FFFF FFFF$_H$	32	FFE6 3294$_H$	1(1H)	32
ATU4F	General Register AF9	GRAF9	FFFF FFFF$_H$	32	FFE6 3294$_H$	1(1H)	32
ATU4F	Event Counter F9	ECNTBF9	0000$_H$	16	FFE6 3298$_H$	1(1H)	16
ATU4F	General Register BF9	GRBF9	FFFF$_H$	16	FFE6 329C$_H$	1(1H)	16
ATU4F	Timer Counter CF9	ECNTCF9	0000 0000$_H$	32	FFE6 32A0$_H$	1(1H)	32
ATU4F	Back-up Register CF9	BGRCF9	FFFF FFFF$_H$	32	FFE6 32A4$_H$	1(1H)	32
ATU4F	General Register CF9	GRCF9	FFFF FFFF$_H$	32	FFE6 32A4$_H$	1(1H)	32
ATU4F	General Register DF9	GRDF9	FFFF FFFF$_H$	32	FFE6 32A8$_H$	1(1H)	32
ATU4F	Back-up Register DF9	BGRDF9	FFFF FFFF$_H$	32	FFE6 32A8$_H$	1(1H)	32
ATU4F	Capture Output Register F9	CDRF9	0000 FFFF$_H$	32	FFE6 32AC$_H$	1(1H)	32
ATU4F	Timer Control Register 1F10	TCR1F10	00$_H$	8	FFE6 32C0$_H$	1(1H)	8
ATU4F	Timer Control Register 2F10	TCR2F10	00$_H$	8	FFE6 32C2$_H$	1(1H)	8
ATU4F	Back-up Control Register F10	BKCRF10	00$_H$	8	FFE6 32C4$_H$	1(1H)	8

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
ATU4F	Timer Status Register F10	TSRF10	00_H	8	FFE6 32C5$_H$	1(1H)	8
ATU4F	Timer Status Clear Register F10	TSCRF10	00_H	8	FFE6 32C6$_H$	1(1H)	8
ATU4F	Timer Interrupt Enable Register F10	TIERF10	00_H	8	FFE6 32C7$_H$	1(1H)	8
ATU4F	Noise Cancel Counter FA10	NCNTFA10	0000_H	16	FFE6 32C8$_H$	1(1H)	16
ATU4F	Noise cancel Register FA10	NCRFA10	0000_H	16	FFE6 32CA$_H$	1(1H)	16
ATU4F	Timer Counter AF10	ECNTAF10	0000 0000_H	32	FFE6 32D0$_H$	1(1H)	32
ATU4F	Back-up Register AF10	BGRAF10	FFFF $FFFF_H$	32	FFE6 32D4$_H$	1(1H)	32
ATU4F	General Register AF10	GRAF10	FFFF $FFFF_H$	32	FFE6 32D4$_H$	1(1H)	32
ATU4F	Event Counter F10	ECNTBF10	0000_H	16	FFE6 32D8$_H$	1(1H)	16
ATU4F	General Register BF10	GRBF10	$FFFF_H$	16	FFE6 32DC$_H$	1(1H)	16
ATU4F	Timer Counter CF10	ECNTCF10	0000 0000_H	32	FFE6 32E0$_H$	1(1H)	32
ATU4F	Back-up Register CF10	BGRCF10	FFFF $FFFF_H$	32	FFE6 32E4$_H$	1(1H)	32
ATU4F	General Register CF10	GRCF10	FFFF $FFFF_H$	32	FFE6 32E4$_H$	1(1H)	32
ATU4F	General Register DF10	GRDF10	FFFF $FFFF_H$	32	FFE6 32E8$_H$	1(1H)	32
ATU4F	Back-up Register DF10	BGRDF10	FFFF $FFFF_H$	32	FFE6 32E8$_H$	1(1H)	32
ATU4F	Capture Output Register F10	CDRF10	0000 $FFFF_H$	32	FFE6 32EC$_H$	1(1H)	32
ATU4F	Timer Control Register 1F11	TCR1F11	00_H	8	FFE6 3300$_H$	1(1H)	8
ATU4F	Timer Control Register 2F11	TCR2F11	00_H	8	FFE6 3302$_H$	1(1H)	8
ATU4F	Back-up Control Register F11	BKCRF11	00_H	8	FFE6 3304$_H$	1(1H)	8
ATU4F	Timer Status Register F11	TSRF11	00_H	8	FFE6 3305$_H$	1(1H)	8
ATU4F	Timer Status Clear Register F11	TSCRF11	00_H	8	FFE6 3306$_H$	1(1H)	8
ATU4F	Timer Interrupt Enable Register F11	TIERF11	00_H	8	FFE6 3307$_H$	1(1H)	8
ATU4F	Noise Cancel Counter FA11	NCNTFA11	0000_H	16	FFE6 3308$_H$	1(1H)	16
ATU4F	Noise cancel Register FA11	NCRFA11	0000_H	16	FFE6 330A$_H$	1(1H)	16
ATU4F	Timer Counter AF11	ECNTAF11	0000 0000_H	32	FFE6 3310$_H$	1(1H)	32
ATU4F	Back-up Register AF11	BGRAF11	FFFF $FFFF_H$	32	FFE6 3314$_H$	1(1H)	32
ATU4F	General Register AF11	GRAF11	FFFF $FFFF_H$	32	FFE6 3314$_H$	1(1H)	32
ATU4F	Event Counter F11	ECNTBF11	0000_H	16	FFE6 3318$_H$	1(1H)	16
ATU4F	General Register BF11	GRBF11	$FFFF_H$	16	FFE6 331C$_H$	1(1H)	16
ATU4F	Timer Counter CF11	ECNTCF11	0000 0000_H	32	FFE6 3320$_H$	1(1H)	32
ATU4F	Back-up Register CF11	BGRCF11	FFFF $FFFF_H$	32	FFE6 3324$_H$	1(1H)	32
ATU4F	General Register CF11	GRCF11	FFFF $FFFF_H$	32	FFE6 3324$_H$	1(1H)	32
ATU4F	General Register DF11	GRDF11	FFFF $FFFF_H$	32	FFE6 3328$_H$	1(1H)	32
ATU4F	Back-up Register DF11	BGRDF11	FFFF $FFFF_H$	32	FFE6 3328$_H$	1(1H)	32
ATU4F	Capture Output Register F11	CDRF11	0000 $FFFF_H$	32	FFE6 332C$_H$	1(1H)	32
ATU4F	Timer Control Register 1F12	TCR1F12	00_H	8	FFE6 3340$_H$	1(1H)	8
ATU4F	Timer Control Register 2F12	TCR2F12	00_H	8	FFE6 3342$_H$	1(1H)	8
ATU4F	Back-up Control Register F12	BKCRF12	00_H	8	FFE6 3344$_H$	1(1H)	8
ATU4F	Timer Status Register F12	TSRF12	00_H	8	FFE6 3345$_H$	1(1H)	8
ATU4F	Timer Status Clear Register F12	TSCRF12	00_H	8	FFE6 3346$_H$	1(1H)	8
ATU4F	Timer Interrupt Enable Register F12	TIERF12	00_H	8	FFE6 3347$_H$	1(1H)	8
ATU4F	Noise Cancel Counter FA12	NCNTFA12	0000_H	16	FFE6 3348$_H$	1(1H)	16
ATU4F	Noise cancel Register FA12	NCRFA12	0000_H	16	FFE6 334A$_H$	1(1H)	16
ATU4F	Timer Counter AF12	ECNTAF12	0000 0000_H	32	FFE6 3350$_H$	1(1H)	32
ATU4F	Back-up Register AF12	BGRAF12	FFFF $FFFF_H$	32	FFE6 3354$_H$	1(1H)	32
ATU4F	General Register AF12	GRAF12	FFFF $FFFF_H$	32	FFE6 3354$_H$	1(1H)	32
ATU4F	Event Counter F12	ECNTBF12	0000_H	16	FFE6 3358$_H$	1(1H)	16

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
ATU4F	General Register BF12	GRBF12	$FFFF_H$	16	$FFE6\ 335C_H$	1(1H)	16
ATU4F	Timer Counter CF12	ECNTCF12	$0000\ 0000_H$	32	$FFE6\ 3360_H$	1(1H)	32
ATU4F	Back-up Register CF12	BGRCF12	$FFFF\ FFFF_H$	32	$FFE6\ 3364_H$	1(1H)	32
ATU4F	General Register CF12	GRCF12	$FFFF\ FFFF_H$	32	$FFE6\ 3364_H$	1(1H)	32
ATU4F	General Register DF12	GRDF12	$FFFF\ FFFF_H$	32	$FFE6\ 3368_H$	1(1H)	32
ATU4F	Back-up Register DF12	BGRDF12	$FFFF\ FFFF_H$	32	$FFE6\ 3368_H$	1(1H)	32
ATU4F	Capture Output Register F12	CDRF12	$0000\ FFFF_H$	32	$FFE6\ 336C_H$	1(1H)	32
ATU4F	Timer Control Register 1F13	TCR1F13	00_H	8	$FFE6\ 3380_H$	1(1H)	8
ATU4F	Timer Control Register 2F13	TCR2F13	00_H	8	$FFE6\ 3382_H$	1(1H)	8
ATU4F	Back-up Control Register F13	BKCRF13	00_H	8	$FFE6\ 3384_H$	1(1H)	8
ATU4F	Timer Status Register F13	TSRF13	00_H	8	$FFE6\ 3385_H$	1(1H)	8
ATU4F	Timer Status Clear Register F13	TSCRF13	00_H	8	$FFE6\ 3386_H$	1(1H)	8
ATU4F	Timer Interrupt Enable Register F13	TIERF13	00_H	8	$FFE6\ 3387_H$	1(1H)	8
ATU4F	Noise Cancel Counter FA13	NCNTFA13	0000_H	16	$FFE6\ 3388_H$	1(1H)	16
ATU4F	Noise cancel Register FA13	NCRFA13	0000_H	16	$FFE6\ 338A_H$	1(1H)	16
ATU4F	Timer Counter AF13	ECNTAF13	$0000\ 0000_H$	32	$FFE6\ 3390_H$	1(1H)	32
ATU4F	Back-up Register AF13	BGRAF13	$FFFF\ FFFF_H$	32	$FFE6\ 3394_H$	1(1H)	32
ATU4F	General Register AF13	GRAF13	$FFFF\ FFFF_H$	32	$FFE6\ 3394_H$	1(1H)	32
ATU4F	Event Counter F13	ECNTBF13	0000_H	16	$FFE6\ 3398_H$	1(1H)	16
ATU4F	General Register BF13	GRBF13	$FFFF_H$	16	$FFE6\ 339C_H$	1(1H)	16
ATU4F	Timer Counter CF13	ECNTCF13	$0000\ 0000_H$	32	$FFE6\ 33A0_H$	1(1H)	32
ATU4F	Back-up Register CF13	BGRCF13	$FFFF\ FFFF_H$	32	$FFE6\ 33A4_H$	1(1H)	32
ATU4F	General Register CF13	GRCF13	$FFFF\ FFFF_H$	32	$FFE6\ 33A4_H$	1(1H)	32
ATU4F	General Register DF13	GRDF13	$FFFF\ FFFF_H$	32	$FFE6\ 33A8_H$	1(1H)	32
ATU4F	Back-up Register DF13	BGRDF13	$FFFF\ FFFF_H$	32	$FFE6\ 33A8_H$	1(1H)	32
ATU4F	Capture Output Register F13	CDRF13	$0000\ FFFF_H$	32	$FFE6\ 33AC_H$	1(1H)	32
ATU4F	Timer Control Register 1F14	TCR1F14	00_H	8	$FFE6\ 33C0_H$	1(1H)	8
ATU4F	Timer Control Register 2F14	TCR2F14	00_H	8	$FFE6\ 33C2_H$	1(1H)	8
ATU4F	Back-up Control Register F14	BKCRF14	00_H	8	$FFE6\ 33C4_H$	1(1H)	8
ATU4F	Timer Status Register F14	TSRF14	00_H	8	$FFE6\ 33C5_H$	1(1H)	8
ATU4F	Timer Status Clear Register F14	TSCRF14	00_H	8	$FFE6\ 33C6_H$	1(1H)	8
ATU4F	Timer Interrupt Enable Register F14	TIERF14	00_H	8	$FFE6\ 33C7_H$	1(1H)	8
ATU4F	Noise Cancel Counter FA14	NCNTFA14	0000_H	16	$FFE6\ 33C8_H$	1(1H)	16
ATU4F	Noise cancel Register FA14	NCRFA14	0000_H	16	$FFE6\ 33CA_H$	1(1H)	16
ATU4F	Timer Counter AF14	ECNTAF14	$0000\ 0000_H$	32	$FFE6\ 33D0_H$	1(1H)	32
ATU4F	Back-up Register AF14	BGRAF14	$FFFF\ FFFF_H$	32	$FFE6\ 33D4_H$	1(1H)	32
ATU4F	General Register AF14	GRAF14	$FFFF\ FFFF_H$	32	$FFE6\ 33D4_H$	1(1H)	32
ATU4F	Event Counter F14	ECNTBF14	0000_H	16	$FFE6\ 33D8_H$	1(1H)	16
ATU4F	General Register BF14	GRBF14	$FFFF_H$	16	$FFE6\ 33DC_H$	1(1H)	16
ATU4F	Timer Counter CF14	ECNTCF14	$0000\ 0000_H$	32	$FFE6\ 33E0_H$	1(1H)	32
ATU4F	Back-up Register CF14	BGRCF14	$FFFF\ FFFF_H$	32	$FFE6\ 33E4_H$	1(1H)	32
ATU4F	General Register CF14	GRCF14	$FFFF\ FFFF_H$	32	$FFE6\ 33E4_H$	1(1H)	32
ATU4F	General Register DF14	GRDF14	$FFFF\ FFFF_H$	32	$FFE6\ 33E8_H$	1(1H)	32
ATU4F	Back-up Register DF14	BGRDF14	$FFFF\ FFFF_H$	32	$FFE6\ 33E8_H$	1(1H)	32
ATU4F	Capture Output Register F14	CDRF14	$0000\ FFFF_H$	32	$FFE6\ 33EC_H$	1(1H)	32
ATU4F	Timer Control Register 1F15	TCR1F15	00_H	8	$FFE6\ 3400_H$	1(1H)	8
ATU4F	Timer Control Register 2F15	TCR2F15	00_H	8	$FFE6\ 3402_H$	1(1H)	8

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
ATU4F	Back-up Control Register F15	BKCRF15	00_H	8	FFE6 3404_H	1(1H)	8
ATU4F	Timer Status Register F15	TSRF15	00_H	8	FFE6 3405_H	1(1H)	8
ATU4F	Timer Status Clear Register F15	TSCRF15	00_H	8	FFE6 3406_H	1(1H)	8
ATU4F	Timer Interrupt Enable Register F15	TIERF15	00_H	8	FFE6 3407_H	1(1H)	8
ATU4F	Noise Cancel Counter FA15	NCNTFA15	0000_H	16	FFE6 3408_H	1(1H)	16
ATU4F	Noise cancel Register FA15	NCRFA15	0000_H	16	FFE6 $340A_H$	1(1H)	16
ATU4F	Timer Counter AF15	ECNTAF15	$0000\ 0000_H$	32	FFE6 3410_H	1(1H)	32
ATU4F	Back-up Register AF15	BGRAF15	$FFFF\ FFFF_H$	32	FFE6 3414_H	1(1H)	32
ATU4F	General Register AF15	GRAF15	$FFFF\ FFFF_H$	32	FFE6 3414_H	1(1H)	32
ATU4F	Event Counter F15	ECNTBF15	0000_H	16	FFE6 3418_H	1(1H)	16
ATU4F	General Register BF15	GRBF15	$FFFF_H$	16	FFE6 $341C_H$	1(1H)	16
ATU4F	Timer Counter CF15	ECNTCF15	$0000\ 0000_H$	32	FFE6 3420_H	1(1H)	32
ATU4F	Back-up Register CF15	BGRCF15	$FFFF\ FFFF_H$	32	FFE6 3424_H	1(1H)	32
ATU4F	General Register CF15	GRCF15	$FFFF\ FFFF_H$	32	FFE6 3424_H	1(1H)	32
ATU4F	General Register DF15	GRDF15	$FFFF\ FFFF_H$	32	FFE6 3428_H	1(1H)	32
ATU4F	Back-up Register DF15	BGRDF15	$FFFF\ FFFF_H$	32	FFE6 3428_H	1(1H)	32
ATU4F	Capture Output Register F15	CDRF15	$0000\ FFFF_H$	32	FFE6 $342C_H$	1(1H)	32
ATU4G	Timer Start Register G	TSTRG	0000_H	16	FFE6 3900_H	1(1H)	8, 16
ATU4G	Timer Control Register G0	TCRG0	00_H	8	FFE6 3910_H	1(1H)	8
ATU4G	Timer Status Register G0	TSRG0	00_H	8	FFE6 3912_H	1(1H)	8
ATU4G	Timer Status Clear Register G0	TSCRG0	00_H	8	FFE6 3913_H	1(1H)	8
ATU4G	Timer counter G0	TCNTG0	$0000\ 0000_H$	32	FFE6 3914_H	1(1H)	32
ATU4G	Compare match Register G0	OCRG0	$FFFF\ FFFF_H$	32	FFE6 3918_H	1(1H)	32
ATU4G	Timer Control Register G1	TCRG1	00_H	8	FFE6 3920_H	1(1H)	8
ATU4G	Timer Status Register G1	TSRG1	00_H	8	FFE6 3922_H	1(1H)	8
ATU4G	Timer Status Clear Register G1	TSCRG1	00_H	8	FFE6 3923_H	1(1H)	8
ATU4G	Timer counter G1	TCNTG1	$0000\ 0000_H$	32	FFE6 3924_H	1(1H)	32
ATU4G	Compare match Register G1	OCRG1	$FFFF\ FFFF_H$	32	FFE6 3928_H	1(1H)	32
ATU4G	Timer Control Register G2	TCRG2	00_H	8	FFE6 3930_H	1(1H)	8
ATU4G	Timer Status Register G2	TSRG2	00_H	8	FFE6 3932_H	1(1H)	8
ATU4G	Timer Status Clear Register G2	TSCRG2	00_H	8	FFE6 3933_H	1(1H)	8
ATU4G	Timer counter G2	TCNTG2	$0000\ 0000_H$	32	FFE6 3934_H	1(1H)	32
ATU4G	Compare match Register G2	OCRG2	$FFFF\ FFFF_H$	32	FFE6 3938_H	1(1H)	32
ATU4G	Timer Control Register G3	TCRG3	00_H	8	FFE6 3940_H	1(1H)	8
ATU4G	Timer Status Register G3	TSRG3	00_H	8	FFE6 3942_H	1(1H)	8
ATU4G	Timer Status Clear Register G3	TSCRG3	00_H	8	FFE6 3943_H	1(1H)	8
ATU4G	Timer counter G3	TCNTG3	$0000\ 0000_H$	32	FFE6 3944_H	1(1H)	32
ATU4G	Compare match Register G3	OCRG3	$FFFF\ FFFF_H$	32	FFE6 3948_H	1(1H)	32
ATU4G	Timer Control Register G4	TCRG4	00_H	8	FFE6 3950_H	1(1H)	8
ATU4G	Timer Status Register G4	TSRG4	00_H	8	FFE6 3952_H	1(1H)	8
ATU4G	Timer Status Clear Register G4	TSCRG4	00_H	8	FFE6 3953_H	1(1H)	8
ATU4G	Timer counter G4	TCNTG4	$0000\ 0000_H$	32	FFE6 3954_H	1(1H)	32
ATU4G	Compare match Register G4	OCRG4	$FFFF\ FFFF_H$	32	FFE6 3958_H	1(1H)	32
ATU4G	Timer Control Register G5	TCRG5	00_H	8	FFE6 3960_H	1(1H)	8
ATU4G	Timer Status Register G5	TSRG5	00_H	8	FFE6 3962_H	1(1H)	8
ATU4G	Timer Status Clear Register G5	TSCRG5	00_H	8	FFE6 3963_H	1(1H)	8
ATU4G	Timer counter G5	TCNTG5	$0000\ 0000_H$	32	FFE6 3964_H	1(1H)	32

(97/201)

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
ATU4G	Compare match Register G5	OCRG5	FFFF FFFF$_H$	32	FFE6 3968$_H$	1(1H)	32
ATU4G	Timer Control Register G6	TCRG6	00$_H$	8	FFE6 3970$_H$	1(1H)	8
ATU4G	Timer Status Register G6	TSRG6	00$_H$	8	FFE6 3972$_H$	1(1H)	8
ATU4G	Timer Status Clear Register G6	TSCRG6	00$_H$	8	FFE6 3973$_H$	1(1H)	8
ATU4G	Timer counter G6	TCNTG6	0000 0000$_H$	32	FFE6 3974$_H$	1(1H)	32
ATU4G	Compare match Register G6	OCRG6	FFFF FFFF$_H$	32	FFE6 3978$_H$	1(1H)	32
ATU4G	Timer Control Register G7	TCRG7	00$_H$	8	FFE6 3980$_H$	1(1H)	8
ATU4G	Timer Status Register G7	TSRG7	00$_H$	8	FFE6 3982$_H$	1(1H)	8
ATU4G	Timer Status Clear Register G7	TSCRG7	00$_H$	8	FFE6 3983$_H$	1(1H)	8
ATU4G	Timer counter G7	TCNTG7	0000 0000$_H$	32	FFE6 3984$_H$	1(1H)	32
ATU4G	Compare match Register G7	OCRG7	FFFF FFFF$_H$	32	FFE6 3988$_H$	1(1H)	32
ATU4G	Timer Control Register G8	TCRG8	00$_H$	8	FFE6 3990$_H$	1(1H)	8
ATU4G	Timer Status Register G8	TSRG8	00$_H$	8	FFE6 3992$_H$	1(1H)	8
ATU4G	Timer Status Clear Register G8	TSCRG8	00$_H$	8	FFE6 3993$_H$	1(1H)	8
ATU4G	Timer counter G8	TCNTG8	0000 0000$_H$	32	FFE6 3994$_H$	1(1H)	32
ATU4G	Compare match Register G8	OCRG8	FFFF FFFF$_H$	32	FFE6 3998$_H$	1(1H)	32
ATU4G	Timer Control Register G9	TCRG9	00$_H$	8	FFE6 39A0$_H$	1(1H)	8
ATU4G	Timer Status Register G9	TSRG9	00$_H$	8	FFE6 39A2$_H$	1(1H)	8
ATU4G	Timer Status Clear Register G9	TSCRG9	00$_H$	8	FFE6 39A3$_H$	1(1H)	8
ATU4G	Timer counter G9	TCNTG9	0000 0000$_H$	32	FFE6 39A4$_H$	1(1H)	32
ATU4G	Compare match Register G9	OCRG9	FFFF FFFF$_H$	32	FFE6 39A8$_H$	1(1H)	32
ATU4J	Timer Start Register J	TSTRJ	00$_H$	8	FFE6 3C00$_H$	1(1H)	8
ATU4J	Timer Control Register J0	TCRJ0	00$_H$	8	FFE6 3C20$_H$	1(1H)	8
ATU4J	FIFO Control Register J0	FCRJ0	00$_H$	8	FFE6 3C21$_H$	1(1H)	8
ATU4J	Timer Status Register J0	TSRJ0	00$_H$	8	FFE6 3C22$_H$	1(1H)	8
ATU4J	Timer Status Clear Register J0	TSCRJ0	00$_H$	8	FFE6 3C23$_H$	1(1H)	8
ATU4J	Timer counter J0	TCNTJ0	0000 0000$_H$	32	FFE6 3C24$_H$	1(1H)	32
ATU4J	Compare match Register J0	OCRJ0	FFFF FFFF$_H$	32	FFE6 3C28$_H$	1(1H)	32
ATU4J	FIFO Register J0	FIFOJ0	0000 0000$_H$	32	FFE6 3C2C$_H$	1(1H)	32
ATU4J	FIFO Data Count Register J0	FDNRJ0	00$_H$	8	FFE6 3C30$_H$	1(1H)	8
ATU4J	Noise Cancel Counter J0	NCNTJ0	0000$_H$	16	FFE6 3C34$_H$	1(1H)	16
ATU4J	Noise cancel Register J0	NCRJ0	0000$_H$	16	FFE6 3C36$_H$	1(1H)	16
ATU4J	Timer Control Register J1	TCRJ1	00$_H$	8	FFE6 3C40$_H$	1(1H)	8
ATU4J	FIFO Control Register J1	FCRJ1	00$_H$	8	FFE6 3C41$_H$	1(1H)	8
ATU4J	Timer Status Register J1	TSRJ1	00$_H$	8	FFE6 3C42$_H$	1(1H)	8
ATU4J	Timer Status Clear Register J1	TSCRJ1	00$_H$	8	FFE6 3C43$_H$	1(1H)	8
ATU4J	Timer counter J1	TCNTJ1	0000 0000$_H$	32	FFE6 3C44$_H$	1(1H)	32
ATU4J	Compare match Register J1	OCRJ1	FFFF FFFF$_H$	32	FFE6 3C48$_H$	1(1H)	32
ATU4J	FIFO Register J1	FIFOJ1	0000 0000$_H$	32	FFE6 3C4C$_H$	1(1H)	32
ATU4J	FIFO Data Count Register J1	FDNRJ1	00$_H$	8	FFE6 3C50$_H$	1(1H)	8
ATU4J	Noise Cancel Counter J1	NCNTJ1	0000$_H$	16	FFE6 3C54$_H$	1(1H)	16
ATU4J	Noise cancel Register J1	NCRJ1	0000$_H$	16	FFE6 3C56$_H$	1(1H)	16
ATU4J	Timer Control Register J2	TCRJ2	00$_H$	8	FFE6 3C60$_H$	1(1H)	8
ATU4J	FIFO Control Register J2	FCRJ2	00$_H$	8	FFE6 3C61$_H$	1(1H)	8
ATU4J	Timer Status Register J2	TSRJ2	00$_H$	8	FFE6 3C62$_H$	1(1H)	8
ATU4J	Timer Status Clear Register J2	TSCRJ2	00$_H$	8	FFE6 3C63$_H$	1(1H)	8
ATU4J	Timer counter J2	TCNTJ2	0000 0000$_H$	32	FFE6 3C64$_H$	1(1H)	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
ATU4J	Compare match Register J2	OCRJ2	FFFF FFFF$_H$	32	FFE6 3C68$_H$	1(1H)	32
ATU4J	FIFO Register J2	FIFOJ2	0000 0000$_H$	32	FFE6 3C6C$_H$	1(1H)	32
ATU4J	FIFO Data Count Register J2	FDNRJ2	00$_H$	8	FFE6 3C70$_H$	1(1H)	8
ATU4J	Noise Cancel Counter J2	NCNTJ2	0000$_H$	16	FFE6 3C74$_H$	1(1H)	16
ATU4J	Noise cancel Register J2	NCRJ2	0000$_H$	16	FFE6 3C76$_H$	1(1H)	16
ATU4J	Timer Control Register J3	TCRJ3	00$_H$	8	FFE6 3C80$_H$	1(1H)	8
ATU4J	FIFO Control Register J3	FCRJ3	00$_H$	8	FFE6 3C81$_H$	1(1H)	8
ATU4J	Timer Status Register J3	TSRJ3	00$_H$	8	FFE6 3C82$_H$	1(1H)	8
ATU4J	Timer Status Clear Register J3	TSCRJ3	00$_H$	8	FFE6 3C83$_H$	1(1H)	8
ATU4J	Timer counter J3	TCNTJ3	0000 0000$_H$	32	FFE6 3C84$_H$	1(1H)	32
ATU4J	Compare match Register J3	OCRJ3	FFFF FFFF$_H$	32	FFE6 3C88$_H$	1(1H)	32
ATU4J	FIFO Register J3	FIFOJ3	0000 0000$_H$	32	FFE6 3C8C$_H$	1(1H)	32
ATU4J	FIFO Data Count Register J3	FDNRJ3	00$_H$	8	FFE6 3C90$_H$	1(1H)	8
ATU4J	Noise Cancel Counter J3	NCNTJ3	0000$_H$	16	FFE6 3C94$_H$	1(1H)	16
ATU4J	Noise cancel Register J3	NCRJ3	0000$_H$	16	FFE6 3C96$_H$	1(1H)	16
ATU4J	Timer Control Register J4	TCRJ4	00$_H$	8	FFE6 3CA0$_H$	1(1H)	8
ATU4J	FIFO Control Register J4	FCRJ4	00$_H$	8	FFE6 3CA1$_H$	1(1H)	8
ATU4J	Timer Status Register J4	TSRJ4	00$_H$	8	FFE6 3CA2$_H$	1(1H)	8
ATU4J	Timer Status Clear Register J4	TSCRJ4	00$_H$	8	FFE6 3CA3$_H$	1(1H)	8
ATU4J	Timer counter J4	TCNTJ4	0000 0000$_H$	32	FFE6 3CA4$_H$	1(1H)	32
ATU4J	Compare match Register J4	OCRJ4	FFFF FFFF$_H$	32	FFE6 3CA8$_H$	1(1H)	32
ATU4J	FIFO Register J4	FIFOJ4	0000 0000$_H$	32	FFE6 3CAC$_H$	1(1H)	32
ATU4J	FIFO Data Count Register J4	FDNRJ4	00$_H$	8	FFE6 3CB0$_H$	1(1H)	8
ATU4J	Noise Cancel Counter J4	NCNTJ4	0000$_H$	16	FFE6 3CB4$_H$	1(1H)	16
ATU4J	Noise cancel Register J4	NCRJ4	0000$_H$	16	FFE6 3CB6$_H$	1(1H)	16
ATU4J	Timer Control Register J5	TCRJ5	00$_H$	8	FFE6 3CC0$_H$	1(1H)	8
ATU4J	FIFO Control Register J5	FCRJ5	00$_H$	8	FFE6 3CC1$_H$	1(1H)	8
ATU4J	Timer Status Register J5	TSRJ5	00$_H$	8	FFE6 3CC2$_H$	1(1H)	8
ATU4J	Timer Status Clear Register J5	TSCRJ5	00$_H$	8	FFE6 3CC3$_H$	1(1H)	8
ATU4J	Timer counter J5	TCNTJ5	0000 0000$_H$	32	FFE6 3CC4$_H$	1(1H)	32
ATU4J	Compare match Register J5	OCRJ5	FFFF FFFF$_H$	32	FFE6 3CC8$_H$	1(1H)	32
ATU4J	FIFO Register J5	FIFOJ5	0000 0000$_H$	32	FFE6 3CCC$_H$	1(1H)	32
ATU4J	FIFO Data Count Register J5	FDNRJ5	00$_H$	8	FFE6 3CD0$_H$	1(1H)	8
ATU4J	Noise Cancel Counter J5	NCNTJ5	0000$_H$	16	FFE6 3CD4$_H$	1(1H)	16
ATU4J	Noise cancel Register J5	NCRJ5	0000$_H$	16	FFE6 3CD6$_H$	1(1H)	16
TSG20	TSG20 I/O Control Register 2	TSG20IOC2	0000$_H$	16	FFE7 0000$_H$	1(1H)	16
TSG20	TSG20 Control Register 3	TSG20CTL3	00$_H$	8	FFE7 0004$_H$	1(1H)	8
TSG20	TSG20 Control Register 5	TSG20CTL5	0000$_H$	16	FFE7 0008$_H$	1(1H)	16
TSG20	TSG20 Control Register 6	TSG20CTL6	0000$_H$	16	FFE7 000C$_H$	1(1H)	16
TSG20	TSG20 Status Register 0	TSG20STR0	00$_H$	8	FFE7 0010$_H$	1(1H)	8
TSG20	TSG20 Status Register 1	TSG20STR1	00$_H$	8	FFE7 0014$_H$	1(1H)	8
TSG20	TSG20 Status Register 2	TSG20STR2	0000$_H$	16	FFE7 0018$_H$	1(1H)	16
TSG20	TSG20 Status Clear Trigger Register	TSG20STC	0000$_H$	16	FFE7 001C$_H$	1(1H)	16
TSG20	TSG20 Option Register 0	TSG20OPT0	00$_H$	8	FFE7 0020$_H$	1(1H)	8
TSG20	TSG20 Option Register 1	TSG20OPT1	00$_H$	8	FFE7 0024$_H$	1(1H)	8
TSG20	TSG20 Counter Register	TSG20CNT	0000$_H$	16	FFE7 0028$_H$	1(1H)	16
TSG20	TSG20 Sub Counter Register	TSG20SBC	0000$_H$	16	FFE7 002C$_H$	1(1H)	16

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
TSG20	TSG20 Trigger Register 0	TSG20TRG0	00_H	8	FFE7 0030_H	1(1H)	8
TSG20	TSG20 Trigger Register 1	TSG20TRG1	00_H	8	FFE7 0034_H	1(1H)	8
TSG20	TSG20 Comparison Register 1,2	TSG20CMP1W	0000 0000_H	32	FFE7 0040_H	1(1H)	32
TSG20	TSG20 Comparison Register 5,6	TSG20CMP5W	0000 0000_H	32	FFE7 0044_H	1(1H)	32
TSG20	TSG20 Comparison Register 9,10	TSG20CMP9W	0000 0000_H	32	FFE7 0048_H	1(1H)	32
TSG20	TSG20 Comparison Register 3,4	TSG20CMP3W	0000 0000_H	32	FFE7 $004C_H$	1(1H)	32
TSG20	TSG20 Comparison Register 7,8	TSG20CMP7W	0000 0000_H	32	FFE7 0050_H	1(1H)	32
TSG20	TSG20 Comparison Register 11,12	TSG20CMP11W	0000 0000_H	32	FFE7 0054_H	1(1H)	32
TSG20	TSG20 Comparison Register 0	TSG20CMP0	0000 0000_H	32	FFE7 0058_H	1(1H)	32
TSG20	TSG20 Diagnostic Comparison Register 0,1	TSG20DCMP0W	0000 0000_H	32	FFE7 $005C_H$	1(1H)	32
TSG20	TSG20 Diagnostic Comparison Register 2	TSG20DCMP2	0000 0000_H	32	FFE7 0060_H	1(1H)	32
TSG20	TSG20 Pattern Register 0	TSG20PAT0W	0000 0000_H	32	FFE7 0064_H	1(1H)	32
TSG20	TSG20 Pattern Register 1	TSG20PAT1W	0000 0000_H	32	FFE7 0068_H	1(1H)	32
TSG20	TSG20 Dead Time Comparison Register 0	TSG20DTC0W	0000 0000_H	32	FFE7 $006C_H$	1(1H)	32
TSG20	TSG20 Dead Time Comparison Register 1	TSG20DTC1W	0000 0000_H	32	FFE7 0070_H	1(1H)	32
TSG20	TSG20 I/O Control Register 3	TSG20IOC3	0000 0000_H	32	FFE7 0074_H	1(1H)	32
TSG20	TSG20 Control Register 4	TSG20CTL4	0000 0000_H	32	FFE7 $007C_H$	1(1H)	32
TSG20	TSG20 Comparison Register 1	TSG20CMP1	0000_H	16	FFE7 0080_H	1(1H)	16
TSG20	TSG20 Comparison Register 2	TSG20CMP2	0000_H	16	FFE7 0084_H	1(1H)	16
TSG20	TSG20 Comparison Register 5	TSG20CMP5	0000_H	16	FFE7 0088_H	1(1H)	16
TSG20	TSG20 Comparison Register 6	TSG20CMP6	0000_H	16	FFE7 $008C_H$	1(1H)	16
TSG20	TSG20 Comparison Register 9	TSG20CMP9	0000_H	16	FFE7 0090_H	1(1H)	16
TSG20	TSG20 Comparison Register 10	TSG20CMP10	0000_H	16	FFE7 0094_H	1(1H)	16
TSG20	TSG20 Comparison Register 3	TSG20CMP3	0000_H	16	FFE7 0098_H	1(1H)	16
TSG20	TSG20 Comparison Register 4	TSG20CMP4	0000_H	16	FFE7 $009C_H$	1(1H)	16
TSG20	TSG20 Comparison Register 7	TSG20CMP7	0000_H	16	FFE7 $00A0_H$	1(1H)	16
TSG20	TSG20 Comparison Register 8	TSG20CMP8	0000_H	16	FFE7 $00A4_H$	1(1H)	16
TSG20	TSG20 Comparison Register 11	TSG20CMP11	0000_H	16	FFE7 $00A8_H$	1(1H)	16
TSG20	TSG20 Comparison Register 12	TSG20CMP12	0000_H	16	FFE7 $00AC_H$	1(1H)	16
TSG20	TSG20 U Phase Comparison Register	TSG20CMPU	0000_H	16	FFE7 $00B0_H$	1(1H)	16
TSG20	TSG20 V Phase Comparison Register	TSG20CMPV	0000_H	16	FFE7 $00B4_H$	1(1H)	16
TSG20	TSG20 W Phase Comparison Register	TSG20CMPW	0000_H	16	FFE7 $00B8_H$	1(1H)	16
TSG20	TSG20 U Phase Period Register	TSG20UPW	0000_H	16	FFE7 $00BC_H$	1(1H)	16
TSG20	TSG20 V Phase Period Register	TSG20VPW	0000_H	16	FFE7 $00C0_H$	1(1H)	16
TSG20	TSG20 W Phase Period Register	TSG20WPW	0000_H	16	FFE7 $00C4_H$	1(1H)	16
TSG20	TSG20 I/O Control Register 0	TSG20IOC0	$7E_H$	8	FFE7 0200_H	1(1H)	8
TSG20	TSG20 I/O Control Register 1	TSG20IOC1	00_H	8	FFE7 0204_H	1(1H)	8
TSG20	TSG20 Control Register 0	TSG20CTL0	00_H	8	FFE7 0208_H	1(1H)	8
TSG20	TSG20 Control Register 1	TSG20CTL1	0000_H	16	FFE7 $020C_H$	1(1H)	16
TSG20	TSG20 Dead Time Protection Register	TSG20DTPR	0000_H	16	FFE7 0210_H	1(1H)	16
TSG21	TSG21 I/O Control Register 2	TSG21IOC2	0000_H	16	FFE7 1000_H	1(1H)	16
TSG21	TSG21 Control Register 3	TSG21CTL3	00_H	8	FFE7 1004_H	1(1H)	8
TSG21	TSG21 Control Register 5	TSG21CTL5	0000_H	16	FFE7 1008_H	1(1H)	16
TSG21	TSG21 Control Register 6	TSG21CTL6	0000_H	16	FFE7 $100C_H$	1(1H)	16
TSG21	TSG21 Status Register 0	TSG21STR0	00_H	8	FFE7 1010_H	1(1H)	8
TSG21	TSG21 Status Register 1	TSG21STR1	00_H	8	FFE7 1014_H	1(1H)	8

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
TSG21	TSG21 Status Register 2	TSG21STR2	0000_H	16	FFE7 1018_H	1(1H)	16
TSG21	TSG21 Status Clear Trigger Register	TSG21STC	0000_H	16	FFE7 $101C_H$	1(1H)	16
TSG21	TSG21 Option Register	TSG21OPT0	00_H	8	FFE7 1020_H	1(1H)	8
TSG21	TSG21 Option Register	TSG21OPT1	00_H	8	FFE7 1024_H	1(1H)	8
TSG21	TSG21 Counter Register	TSG21CNT	0000_H	16	FFE7 1028_H	1(1H)	16
TSG21	TSG21 Sub-Counter Register	TSG21SBC	0000_H	16	FFE7 $102C_H$	1(1H)	16
TSG21	TSG21 Trigger Register 0	TSG21TRG0	00_H	8	FFE7 1030_H	1(1H)	8
TSG21	TSG21 Trigger Register 1	TSG21TRG1	00_H	8	FFE7 1034_H	1(1H)	8
TSG21	TSG21 Comparison Register 1,2	TSG21CMP1W	0000 0000_H	32	FFE7 1040_H	1(1H)	32
TSG21	TSG21 Comparison Register 5,6	TSG21CMP5W	0000 0000_H	32	FFE7 1044_H	1(1H)	32
TSG21	TSG21 Comparison Register 9,10	TSG21CMP9W	0000 0000_H	32	FFE7 1048_H	1(1H)	32
TSG21	TSG21 Comparison Register 3,4	TSG21CMP3W	0000 0000_H	32	FFE7 $104C_H$	1(1H)	32
TSG21	TSG21 Comparison Register 7,8	TSG21CMP7W	0000 0000_H	32	FFE7 1050_H	1(1H)	32
TSG21	TSG21 Comparison Register 11,12	TSG21CMP11W	0000 0000_H	32	FFE7 1054_H	1(1H)	32
TSG21	TSG21 Comparison Register 0	TSG21CMP0	0000 0000_H	32	FFE7 1058_H	1(1H)	32
TSG21	TSG21 Diagnostic Comparison Register 0,1	TSG21DCMP0W	0000 0000_H	32	FFE7 $105C_H$	1(1H)	32
TSG21	TSG21 Diagnostic Comparison Register 2	TSG21DCMP2	0000 0000_H	32	FFE7 1060_H	1(1H)	32
TSG21	TSG21 Pattern Register 0	TSG21PAT0W	0000 0000_H	32	FFE7 1064_H	1(1H)	32
TSG21	TSG21 Pattern Register 1	TSG21PAT1W	0000 0000_H	32	FFE7 1068_H	1(1H)	32
TSG21	TSG21 Dead Time Comparison Register 0	TSG21DTC0W	0000 0000_H	32	FFE7 $106C_H$	1(1H)	32
TSG21	TSG21 Dead Time Comparison Register 1	TSG21DTC1W	0000 0000_H	32	FFE7 1070_H	1(1H)	32
TSG21	TSG21 I/O Control Register 3	TSG21IOC3	0000 0000_H	32	FFE7 1074_H	1(1H)	32
TSG21	TSG21 Control Register 4	TSG21CTL4	0000 0000_H	32	FFE7 $107C_H$	1(1H)	32
TSG21	TSG21 Comparison Register 1	TSG21CMP1	0000_H	16	FFE7 1080_H	1(1H)	16
TSG21	TSG21 Comparison Register 2	TSG21CMP2	0000_H	16	FFE7 1084_H	1(1H)	16
TSG21	TSG21 Comparison Register 5	TSG21CMP5	0000_H	16	FFE7 1088_H	1(1H)	16
TSG21	TSG21 Comparison Register 6	TSG21CMP6	0000_H	16	FFE7 $108C_H$	1(1H)	16
TSG21	TSG21 Comparison Register 9	TSG21CMP9	0000_H	16	FFE7 1090_H	1(1H)	16
TSG21	TSG21 Comparison Register 10	TSG21CMP10	0000_H	16	FFE7 1094_H	1(1H)	16
TSG21	TSG21 Comparison Register 3	TSG21CMP3	0000_H	16	FFE7 1098_H	1(1H)	16
TSG21	TSG21 Comparison Register 4	TSG21CMP4	0000_H	16	FFE7 $109C_H$	1(1H)	16
TSG21	TSG21 Comparison Register 7	TSG21CMP7	0000_H	16	FFE7 $10A0_H$	1(1H)	16
TSG21	TSG21 Comparison Register 8	TSG21CMP8	0000_H	16	FFE7 $10A4_H$	1(1H)	16
TSG21	TSG21 Comparison Register 11	TSG21CMP11	0000_H	16	FFE7 $10A8_H$	1(1H)	16
TSG21	TSG21 Comparison Register 12	TSG21CMP12	0000_H	16	FFE7 $10AC_H$	1(1H)	16
TSG21	TSG21 U Phase Comparison Register	TSG21CMPU	0000_H	16	FFE7 $10B0_H$	1(1H)	16
TSG21	TSG21 V Phase Comparison Register	TSG21CMPV	0000_H	16	FFE7 $10B4_H$	1(1H)	16
TSG21	TSG21 W Phase Comparison Register	TSG21CMPW	0000_H	16	FFE7 $10B8_H$	1(1H)	16
TSG21	TSG21 U Phase Period Register	TSG21UPW	0000_H	16	FFE7 $10BC_H$	1(1H)	16
TSG21	TSG21 V Phase Period Register	TSG21VPW	0000_H	16	FFE7 $10C0_H$	1(1H)	16
TSG21	TSG21 W Phase Period Register	TSG21WPW	0000_H	16	FFE7 $10C4_H$	1(1H)	16
TSG21	TSG21 I/O Control Register 0	TSG21IOC0	$7E_H$	8	FFE7 1200_H	1(1H)	8
TSG21	TSG21 I/O Control Register 1	TSG21IOC1	00_H	8	FFE7 1204_H	1(1H)	8
TSG21	TSG21 Control Register 0	TSG21CTL0	00_H	8	FFE7 1208_H	1(1H)	8
TSG21	TSG21 Control Register 1	TSG21CTL1	0000_H	16	FFE7 $120C_H$	1(1H)	16
TSG21	TSG21 Dead Time Protection Register	TSG21DTPR	0000_H	16	FFE7 1210_H	1(1H)	16

(101/201)

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
TAPA2	TAPA2 Flag Register	TAPA2FLG	0000_H	16	FFE9 2000_H	1(1H)	16
TAPA2	TAPA2 Asynchronous Control Write Enable Register	TAPA2ACWE	00_H	8	FFE9 2004_H	1(1H)	8
TAPA2	TAPA2 Asynchronous Control Start Trigger Register	TAPA2ACTS	00_H	8	FFE9 2008_H	1(1H)	8
TAPA2	TAPA2 Asynchronous Control Stop Trigger Register	TAPA2ACTT	00_H	8	FFE9 $200C_H$	1(1H)	8
TAPA2	TAPA2 Hi-Z Start Trigger Register	TAPA2OPHS	00_H	8	FFE9 2014_H	1(1H)	8
TAPA2	TAPA2 Hi-Z Stop Trigger Register	TAPA2OPHT	00_H	8	FFE9 2018_H	1(1H)	8
TAPA2	TAPA2 Control Register 0	TAPA2CTL0	0000_H	16	FFE9 2020_H	1(1H)	16
TAPA3	TAPA3 Flag Register	TAPA3FLG	0000_H	16	FFE9 3000_H	1(1H)	16
TAPA3	TAPA3 Asynchronous Control Write Enable Register	TAPA3ACWE	00_H	8	FFE9 3004_H	1(1H)	8
TAPA3	TAPA3 Asynchronous Control Start Trigger Register	TAPA3ACTS	00_H	8	FFE9 3008_H	1(1H)	8
TAPA3	TAPA3 Asynchronous Control Stop Trigger Register	TAPA3ACTT	00_H	8	FFE9 $300C_H$	1(1H)	8
TAPA3	TAPA3 Hi-Z Start Trigger Register	TAPA3OPHS	00_H	8	FFE9 3014_H	1(1H)	8
TAPA3	TAPA3 Hi-Z Stop Trigger Register	TAPA3OPHT	00_H	8	FFE9 3018_H	1(1H)	8
TAPA3	TAPA3 Control Register 0	TAPA3CTL0	0000_H	16	FFE9 3020_H	1(1H)	16
APAA0	APAA0 Operation Enable Register	APAA0EN	00_H	8	FFEB 0000_H	1(1L)	8
APAA0	APAA0 Channel Operation Enable Register	APAA0CHEN	0000 0000_H	32	FFEB 0008_H	1(1L)	32
APAA0	APAA0 Channel Output Status Register	APAA0CHST	0000_H	16	FFEB $000C_H$	1(1L)	16
APAA0	APAA0 Reference Data Register 0	APAA0RFDT0	0000_H	16	FFEB 0200_H	1(1L)	16
APAA0	APAA0 Reference Data Register 1	APAA0RFDT1	0000_H	16	FFEB 0204_H	1(1L)	16
APAA0	APAA0 Reference Data Register 2	APAA0RFDT2	0000_H	16	FFEB 0208_H	1(1L)	16
APAA0	APAA0 Reference Data Register 3	APAA0RFDT3	0000_H	16	FFEB $020C_H$	1(1L)	16
APAA0	APAA0 Reference Data Register 4	APAA0RFDT4	0000_H	16	FFEB 0210_H	1(1L)	16
APAA0	APAA0 Reference Data Register 5	APAA0RFDT5	0000_H	16	FFEB 0214_H	1(1L)	16
APAA0	APAA0 Reference Data Register 6	APAA0RFDT6	0000_H	16	FFEB 0218_H	1(1L)	16
APAA0	APAA0 Reference Data Register 7	APAA0RFDT7	0000_H	16	FFEB $021C_H$	1(1L)	16
APAA0	APAA0 Reference Data Register 8	APAA0RFDT8	0000_H	16	FFEB 0220_H	1(1L)	16
APAA0	APAA0 Reference Data Register 9	APAA0RFDT9	0000_H	16	FFEB 0224_H	1(1L)	16
APAA0	APAA0 Software Reference Data Register	APAA0RFSW	0000_H	16	FFEB 0228_H	1(1L)	16
APAA0	APAA0 Reference Maximum Value Setting Register 0	APAA0RFMX0	$FFFF_H$	16	FFEB 0240_H	1(1L)	16
APAA0	APAA0 Reference Maximum Value Setting Register 1	APAA0RFMX1	$FFFF_H$	16	FFEB 0244_H	1(1L)	16
APAA0	APAA0 Software Reference Maximum Value Setting Register	APAA0RFMXSW	$FFFF_H$	16	FFEB 0268_H	1(1L)	16
APAA0	APAA0 Event Selection Register 0	APAA0EVSL00	00_H	8	FFEB 0400_H	1(1L)	8
APAA0	APAA0 Event Selection Register 1	APAA0EVSL01	00_H	8	FFEB 0404_H	1(1L)	8
APAA0	APAA0 Event Selection Register 2	APAA0EVSL02	00_H	8	FFEB 0408_H	1(1L)	8
APAA0	APAA0 Event Selection Register 3	APAA0EVSL03	00_H	8	FFEB $040C_H$	1(1L)	8
APAA0	APAA0 Event Selection Register 4	APAA0EVSL04	00_H	8	FFEB 0410_H	1(1L)	8
APAA0	APAA0 Event Selection Register 5	APAA0EVSL05	00_H	8	FFEB 0414_H	1(1L)	8
APAA0	APAA0 Event Selection Register 6	APAA0EVSL06	00_H	8	FFEB 0418_H	1(1L)	8
APAA0	APAA0 EventS election Register 7	APAA0EVSL07	00_H	8	FFEB $041C_H$	1(1L)	8
APAA0	APAA0 Event Selection Register 8	APAA0EVSL08	00_H	8	FFEB 0420_H	1(1L)	8

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
APAA0	APAA0 Event Selection Register 9	APAA0EVSL09	00_H	8	FFEB 0424_H	1(1L)	8
APAA0	APAA0 Event Selection Register 10	APAA0EVSL10	00_H	8	FFEB 0428_H	1(1L)	8
APAA0	APAA0 Event Selection Register 11	APAA0EVSL11	00_H	8	FFEB $042C_H$	1(1L)	8
APAA0	APAA0 Event Selection Register 12	APAA0EVSL12	00_H	8	FFEB 0430_H	1(1L)	8
APAA0	APAA0 Event Selection Register 13	APAA0EVSL13	00_H	8	FFEB 0434_H	1(1L)	8
APAA0	APAA0 Event Selection Register 14	APAA0EVSL14	00_H	8	FFEB 0438_H	1(1L)	8
APAA0	APAA0 Event Selection Register 15	APAA0EVSL15	00_H	8	FFEB $043C_H$	1(1L)	8
APAA0	APAA0 Software Event Register	APAA0EVSW	$0000\ 0000_H$	32	FFEB 0440_H	1(1L)	32
APAA0	APAA0 Software Event Setting Register	APAA0EVSC	0000_H	16	FFEB 0444_H	1(1L)	16
APAA0	APAA0 Event Status Register A	APAA0ESTA	$0000\ 0000_H$	32	FFEB 0448_H	1(1L)	32
APAA0	APAA0 Channel Setting Register A0	APAA0CCGA00	$0000\ 0100_H$	32	FFEB 0600_H	1(1L)	32
APAA0	APAA0 Channel Setting Register B0	APAA0CCGB00	$0000\ 0000_H$	32	FFEB 0604_H	1(1L)	32
APAA0	APAA0 Channel Status Register A0	APAA0CSTA00	$0000\ 0000_H$	32	FFEB 0608_H	1(1L)	32
APAA0	APAA0 Channel Status Register B0	APAA0CSTB00	$0000\ 0000_H$	32	FFEB $060C_H$	1(1L)	32
APAA0	APAA0 Channel Status Register C0	APAA0CSTC00	$0000\ 0000_H$	32	FFEB 0610_H	1(1L)	32
APAA0	APAA0 Channel Status Register D0	APAA0CSTD00	$0000\ 0000_H$	32	FFEB 0614_H	1(1L)	32
APAA0	APAA0 Channel Setting Register A1	APAA0CCGA01	$0000\ 0100_H$	32	FFEB 0620_H	1(1L)	32
APAA0	APAA0 Channel Setting Register B1	APAA0CCGB01	$0000\ 0000_H$	32	FFEB 0624_H	1(1L)	32
APAA0	APAA0 Channel Status Register A1	APAA0CSTA01	$0000\ 0000_H$	32	FFEB 0628_H	1(1L)	32
APAA0	APAA0 Channel Status Register B1	APAA0CSTB01	$0000\ 0000_H$	32	FFEB $062C_H$	1(1L)	32
APAA0	APAA0 Channel Status Register C1	APAA0CSTC01	$0000\ 0000_H$	32	FFEB 0630_H	1(1L)	32
APAA0	APAA0 Channel Status Register D1	APAA0CSTD01	$0000\ 0000_H$	32	FFEB 0634_H	1(1L)	32
APAA0	APAA0 Channel Setting Register A2	APAA0CCGA02	$0000\ 0100_H$	32	FFEB 0640_H	1(1L)	32
APAA0	APAA0 Channel Setting Register B2	APAA0CCGB02	$0000\ 0000_H$	32	FFEB 0644_H	1(1L)	32
APAA0	APAA0 Channel Status Register A2	APAA0CSTA02	$0000\ 0000_H$	32	FFEB 0648_H	1(1L)	32
APAA0	APAA0 Channel Status Register B2	APAA0CSTB02	$0000\ 0000_H$	32	FFEB $064C_H$	1(1L)	32
APAA0	APAA0 Channel Status Register C2	APAA0CSTC02	$0000\ 0000_H$	32	FFEB 0650_H	1(1L)	32
APAA0	APAA0 Channel Status Register D2	APAA0CSTD02	$0000\ 0000_H$	32	FFEB 0654_H	1(1L)	32
APAA0	APAA0 Channel Setting Register A3	APAA0CCGA03	$0000\ 0100_H$	32	FFEB 0660_H	1(1L)	32
APAA0	APAA0 Channel Setting Register B3	APAA0CCGB03	$0000\ 0000_H$	32	FFEB 0664_H	1(1L)	32
APAA0	APAA0 Channel Status Register A3	APAA0CSTA03	$0000\ 0000_H$	32	FFEB 0668_H	1(1L)	32
APAA0	APAA0 Channel Status Register B3	APAA0CSTB03	$0000\ 0000_H$	32	FFEB $066C_H$	1(1L)	32
APAA0	APAA0 Channel Status Register C3	APAA0CSTC03	$0000\ 0000_H$	32	FFEB 0670_H	1(1L)	32
APAA0	APAA0 Channel Status Register D3	APAA0CSTD03	$0000\ 0000_H$	32	FFEB 0674_H	1(1L)	32
APAA0	APAA0 Channel Setting Register A4	APAA0CCGA04	$0000\ 0100_H$	32	FFEB 0680_H	1(1L)	32
APAA0	APAA0 Channel Setting Register B4	APAA0CCGB04	$0000\ 0000_H$	32	FFEB 0684_H	1(1L)	32
APAA0	APAA0 Channel Status Register A4	APAA0CSTA04	$0000\ 0000_H$	32	FFEB 0688_H	1(1L)	32
APAA0	APAA0 Channel Status Register B4	APAA0CSTB04	$0000\ 0000_H$	32	FFEB $068C_H$	1(1L)	32
APAA0	APAA0 Channel Status Register C4	APAA0CSTC04	$0000\ 0000_H$	32	FFEB 0690_H	1(1L)	32
APAA0	APAA0 Channel Status Register D4	APAA0CSTD04	$0000\ 0000_H$	32	FFEB 0694_H	1(1L)	32
APAA0	APAA0 Channel Setting Register A5	APAA0CCGA05	$0000\ 0100_H$	32	FFEB $06A0_H$	1(1L)	32
APAA0	APAA0 Channel Setting Register B5	APAA0CCGB05	$0000\ 0000_H$	32	FFEB $06A4_H$	1(1L)	32
APAA0	APAA0 Channel Status Register A5	APAA0CSTA05	$0000\ 0000_H$	32	FFEB $06A8_H$	1(1L)	32
APAA0	APAA0 Channel Status Register B5	APAA0CSTB05	$0000\ 0000_H$	32	FFEB $06AC_H$	1(1L)	32
APAA0	APAA0 Channel Status Register C5	APAA0CSTC05	$0000\ 0000_H$	32	FFEB $06B0_H$	1(1L)	32
APAA0	APAA0 Channel Status Register D5	APAA0CSTD05	$0000\ 0000_H$	32	FFEB $06B4_H$	1(1L)	32
APAA0	APAA0 Channel Setting Register A6	APAA0CCGA06	$0000\ 0100_H$	32	FFEB $06C0_H$	1(1L)	32

(103/201)

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
APAA0	APAA0 Channel Setting Register B6	APAA0CCGB06	0000 0000$_H$	32	FFEB 06C4$_H$	1(1L)	32
APAA0	APAA0 Channel Status Register A6	APAA0CSTA06	0000 0000$_H$	32	FFEB 06C8$_H$	1(1L)	32
APAA0	APAA0 Channel Status Register B6	APAA0CSTB06	0000 0000$_H$	32	FFEB 06CC$_H$	1(1L)	32
APAA0	APAA0 Channel Status Register C6	APAA0CSTC06	0000 0000$_H$	32	FFEB 06D0$_H$	1(1L)	32
APAA0	APAA0 Channel Status Register D6	APAA0CSTD06	0000 0000$_H$	32	FFEB 06D4$_H$	1(1L)	32
APAA0	APAA0 Channel Setting Register A7	APAA0CCGA07	0000 0100$_H$	32	FFEB 06E0$_H$	1(1L)	32
APAA0	APAA0 Channel Setting Register B7	APAA0CCGB07	0000 0000$_H$	32	FFEB 06E4$_H$	1(1L)	32
APAA0	APAA0 Channel Status Register A7	APAA0CSTA07	0000 0000$_H$	32	FFEB 06E8$_H$	1(1L)	32
APAA0	APAA0 Channel Status Register B7	APAA0CSTB07	0000 0000$_H$	32	FFEB 06EC$_H$	1(1L)	32
APAA0	APAA0 Channel Status Register C7	APAA0CSTC07	0000 0000$_H$	32	FFEB 06F0$_H$	1(1L)	32
APAA0	APAA0 Channel Status Register D7	APAA0CSTD07	0000 0000$_H$	32	FFEB 06F4$_H$	1(1L)	32
APAA0	APAA0 Channel Setting Register A8	APAA0CCGA08	0000 0100$_H$	32	FFEB 0700$_H$	1(1L)	32
APAA0	APAA0 Channel Setting Register B8	APAA0CCGB08	0000 0000$_H$	32	FFEB 0704$_H$	1(1L)	32
APAA0	APAA0 Channel Status Register A8	APAA0CSTA08	0000 0000$_H$	32	FFEB 0708$_H$	1(1L)	32
APAA0	APAA0 Channel Status Register B8	APAA0CSTB08	0000 0000$_H$	32	FFEB 070C$_H$	1(1L)	32
APAA0	APAA0 Channel Status Register C8	APAA0CSTC08	0000 0000$_H$	32	FFEB 0710$_H$	1(1L)	32
APAA0	APAA0 Channel Status Register D8	APAA0CSTD08	0000 0000$_H$	32	FFEB 0714$_H$	1(1L)	32
APAA0	APAA0 Channel Setting Register A9	APAA0CCGA09	0000 0100$_H$	32	FFEB 0720$_H$	1(1L)	32
APAA0	APAA0 Channel Setting Register B9	APAA0CCGB09	0000 0000$_H$	32	FFEB 0724$_H$	1(1L)	32
APAA0	APAA0 Channel Status Register A9	APAA0CSTA09	0000 0000$_H$	32	FFEB 0728$_H$	1(1L)	32
APAA0	APAA0 Channel Status Register B9	APAA0CSTB09	0000 0000$_H$	32	FFEB 072C$_H$	1(1L)	32
APAA0	APAA0 Channel Status Register C9	APAA0CSTC09	0000 0000$_H$	32	FFEB 0730$_H$	1(1L)	32
APAA0	APAA0 Channel Status Register D9	APAA0CSTD09	0000 0000$_H$	32	FFEB 0734$_H$	1(1L)	32
APAA0	APAA0 Channel Setting Register A10	APAA0CCGA10	0000 0100$_H$	32	FFEB 0740$_H$	1(1L)	32
APAA0	APAA0 Channel Setting Register B10	APAA0CCGB10	0000 0000$_H$	32	FFEB 0744$_H$	1(1L)	32
APAA0	APAA0 Channel Status Register A10	APAA0CSTA10	0000 0000$_H$	32	FFEB 0748$_H$	1(1L)	32
APAA0	APAA0 Channel Status Register B10	APAA0CSTB10	0000 0000$_H$	32	FFEB 074C$_H$	1(1L)	32
APAA0	APAA0 Channel Status Register C10	APAA0CSTC10	0000 0000$_H$	32	FFEB 0750$_H$	1(1L)	32
APAA0	APAA0 Channel Status Register D10	APAA0CSTD10	0000 0000$_H$	32	FFEB 0754$_H$	1(1L)	32
APAA0	APAA0 Channel Setting Register A11	APAA0CCGA11	0000 0100$_H$	32	FFEB 0760$_H$	1(1L)	32
APAA0	APAA0 Channel Setting Register B11	APAA0CCGB11	0000 0000$_H$	32	FFEB 0764$_H$	1(1L)	32
APAA0	APAA0 Channel Status Register A11	APAA0CSTA11	0000 0000$_H$	32	FFEB 0768$_H$	1(1L)	32
APAA0	APAA0 Channel Status Register B11	APAA0CSTB11	0000 0000$_H$	32	FFEB 076C$_H$	1(1L)	32
APAA0	APAA0 Channel Status Register C11	APAA0CSTC11	0000 0000$_H$	32	FFEB 0770$_H$	1(1L)	32
APAA0	APAA0 Channel Status Register D11	APAA0CSTD11	0000 0000$_H$	32	FFEB 0774$_H$	1(1L)	32
APAA0	APAA0 Channel Setting Register A12	APAA0CCGA12	0000 0100$_H$	32	FFEB 0780$_H$	1(1L)	32
APAA0	APAA0 Channel Setting Register B12	APAA0CCGB12	0000 0000$_H$	32	FFEB 0784$_H$	1(1L)	32
APAA0	APAA0 Channel Status Register A12	APAA0CSTA12	0000 0000$_H$	32	FFEB 0788$_H$	1(1L)	32
APAA0	APAA0 Channel Status Register B12	APAA0CSTB12	0000 0000$_H$	32	FFEB 078C$_H$	1(1L)	32
APAA0	APAA0 Channel Status Register C12	APAA0CSTC12	0000 0000$_H$	32	FFEB 0790$_H$	1(1L)	32
APAA0	APAA0 Channel Status Register D12	APAA0CSTD12	0000 0000$_H$	32	FFEB 0794$_H$	1(1L)	32
APAA0	APAA0 Channel Setting Register A13	APAA0CCGA13	0000 0100$_H$	32	FFEB 07A0$_H$	1(1L)	32
APAA0	APAA0 Channel Setting Register B13	APAA0CCGB13	0000 0000$_H$	32	FFEB 07A4$_H$	1(1L)	32
APAA0	APAA0 Channel Status Register A13	APAA0CSTA13	0000 0000$_H$	32	FFEB 07A8$_H$	1(1L)	32
APAA0	APAA0 Channel Status Register B13	APAA0CSTB13	0000 0000$_H$	32	FFEB 07AC$_H$	1(1L)	32
APAA0	APAA0 Channel Status Register C13	APAA0CSTC13	0000 0000$_H$	32	FFEB 07B0$_H$	1(1L)	32
APAA0	APAA0 Channel Status Register D13	APAA0CSTD13	0000 0000$_H$	32	FFEB 07B4$_H$	1(1L)	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
APAA0	APAA0 Channel Setting Register A14	APAA0CCGA14	0000 0100_H	32	FFEB $07C0_H$	1(1L)	32
APAA0	APAA0 Channel Setting Register B14	APAA0CCGB14	0000 0000_H	32	FFEB $07C4_H$	1(1L)	32
APAA0	APAA0 Channel Status Register A14	APAA0CSTA14	0000 0000_H	32	FFEB $07C8_H$	1(1L)	32
APAA0	APAA0 Channel Status Register B14	APAA0CSTB14	0000 0000_H	32	FFEB $07CC_H$	1(1L)	32
APAA0	APAA0 Channel Status Register C14	APAA0CSTC14	0000 0000_H	32	FFEB $07D0_H$	1(1L)	32
APAA0	APAA0 Channel Status Register D14	APAA0CSTD14	0000 0000_H	32	FFEB $07D4_H$	1(1L)	32
APAA0	APAA0 Channel Setting Register A15	APAA0CCGA15	0000 0100_H	32	FFEB $07E0_H$	1(1L)	32
APAA0	APAA0 Channel Setting Register B15	APAA0CCGB15	0000 0000_H	32	FFEB $07E4_H$	1(1L)	32
APAA0	APAA0 Channel Status Register A15	APAA0CSTA15	0000 0000_H	32	FFEB $07E8_H$	1(1L)	32
APAA0	APAA0 Channel Status Register B15	APAA0CSTB15	0000 0000_H	32	FFEB $07EC_H$	1(1L)	32
APAA0	APAA0 Channel Status Register C15	APAA0CSTC15	0000 0000_H	32	FFEB $07F0_H$	1(1L)	32
APAA0	APAA0 Channel Status Register D15	APAA0CSTD15	0000 0000_H	32	FFEB $07F4_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A0	APAA0ELMA000	Undefined	32	FFEB 1000_H	1(1L)	32
APAA0	APAA0 Element Setting Register B0	APAA0ELMB000	Undefined	32	FFEB 1004_H	1(1L)	32
APAA0	APAA0 Element Setting Register C0	APAA0ELMC000	Undefined	32	FFEB 1008_H	1(1L)	32
APAA0	APAA0 Element Setting Register A1	APAA0ELMA001	Undefined	32	FFEB 1010_H	1(1L)	32
APAA0	APAA0 Element Setting Register B1	APAA0ELMB001	Undefined	32	FFEB 1014_H	1(1L)	32
APAA0	APAA0 Element Setting Register C1	APAA0ELMC001	Undefined	32	FFEB 1018_H	1(1L)	32
APAA0	APAA0 Element Setting Register A2	APAA0ELMA002	Undefined	32	FFEB 1020_H	1(1L)	32
APAA0	APAA0 Element Setting Register B2	APAA0ELMB002	Undefined	32	FFEB 1024_H	1(1L)	32
APAA0	APAA0 Element Setting Register C2	APAA0ELMC002	Undefined	32	FFEB 1028_H	1(1L)	32
APAA0	APAA0 Element Setting Register A3	APAA0ELMA003	Undefined	32	FFEB 1030_H	1(1L)	32
APAA0	APAA0 Element Setting Register B3	APAA0ELMB003	Undefined	32	FFEB 1034_H	1(1L)	32
APAA0	APAA0 Element Setting Register C3	APAA0ELMC003	Undefined	32	FFEB 1038_H	1(1L)	32
APAA0	APAA0 Element Setting Register A4	APAA0ELMA004	Undefined	32	FFEB 1040_H	1(1L)	32
APAA0	APAA0 Element Setting Register B4	APAA0ELMB004	Undefined	32	FFEB 1044_H	1(1L)	32
APAA0	APAA0 Element Setting Register C4	APAA0ELMC004	Undefined	32	FFEB 1048_H	1(1L)	32
APAA0	APAA0 Element Setting Register A5	APAA0ELMA005	Undefined	32	FFEB 1050_H	1(1L)	32
APAA0	APAA0 Element Setting Register B5	APAA0ELMB005	Undefined	32	FFEB 1054_H	1(1L)	32
APAA0	APAA0 Element Setting Register C5	APAA0ELMC005	Undefined	32	FFEB 1058_H	1(1L)	32
APAA0	APAA0 Element Setting Register A6	APAA0ELMA006	Undefined	32	FFEB 1060_H	1(1L)	32
APAA0	APAA0 Element Setting Register B6	APAA0ELMB006	Undefined	32	FFEB 1064_H	1(1L)	32
APAA0	APAA0 Element Setting Register C6	APAA0ELMC006	Undefined	32	FFEB 1068_H	1(1L)	32
APAA0	APAA0 Element Setting Register A7	APAA0ELMA007	Undefined	32	FFEB 1070_H	1(1L)	32
APAA0	APAA0 Element Setting Register B7	APAA0ELMB007	Undefined	32	FFEB 1074_H	1(1L)	32
APAA0	APAA0 Element Setting Register C7	APAA0ELMC007	Undefined	32	FFEB 1078_H	1(1L)	32
APAA0	APAA0 Element Setting Register A8	APAA0ELMA008	Undefined	32	FFEB 1080_H	1(1L)	32
APAA0	APAA0 Element Setting Register B8	APAA0ELMB008	Undefined	32	FFEB 1084_H	1(1L)	32
APAA0	APAA0 Element Setting Register C8	APAA0ELMC008	Undefined	32	FFEB 1088_H	1(1L)	32
APAA0	APAA0 Element Setting Register A9	APAA0ELMA009	Undefined	32	FFEB 1090_H	1(1L)	32
APAA0	APAA0 Element Setting Register B9	APAA0ELMB009	Undefined	32	FFEB 1094_H	1(1L)	32
APAA0	APAA0 Element Setting Register C9	APAA0ELMC009	Undefined	32	FFEB 1098_H	1(1L)	32
APAA0	APAA0 Element Setting Register A10	APAA0ELMA010	Undefined	32	FFEB $10A0_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B10	APAA0ELMB010	Undefined	32	FFEB $10A4_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C10	APAA0ELMC010	Undefined	32	FFEB $10A8_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A11	APAA0ELMA011	Undefined	32	FFEB $10B0_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B11	APAA0ELMB011	Undefined	32	FFEB $10B4_H$	1(1L)	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
APAA0	APAA0 Element Setting Register C11	APAA0ELMC011	Undefined	32	FFEB 10B8$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A12	APAA0ELMA012	Undefined	32	FFEB 10C0$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B12	APAA0ELMB012	Undefined	32	FFEB 10C4$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C12	APAA0ELMC012	Undefined	32	FFEB 10C8$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A13	APAA0ELMA013	Undefined	32	FFEB 10D0$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B13	APAA0ELMB013	Undefined	32	FFEB 10D4$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C13	APAA0ELMC013	Undefined	32	FFEB 10D8$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A14	APAA0ELMA014	Undefined	32	FFEB 10E0$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B14	APAA0ELMB014	Undefined	32	FFEB 10E4$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C14	APAA0ELMC014	Undefined	32	FFEB 10E8$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A15	APAA0ELMA015	Undefined	32	FFEB 10F0$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B15	APAA0ELMB015	Undefined	32	FFEB 10F4$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C15	APAA0ELMC015	Undefined	32	FFEB 10F8$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A16	APAA0ELMA016	Undefined	32	FFEB 1100$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B16	APAA0ELMB016	Undefined	32	FFEB 1104$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C16	APAA0ELMC016	Undefined	32	FFEB 1108$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A17	APAA0ELMA017	Undefined	32	FFEB 1110$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B17	APAA0ELMB017	Undefined	32	FFEB 1114$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C17	APAA0ELMC017	Undefined	32	FFEB 1118$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A18	APAA0ELMA018	Undefined	32	FFEB 1120$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B18	APAA0ELMB018	Undefined	32	FFEB 1124$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C18	APAA0ELMC018	Undefined	32	FFEB 1128$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A19	APAA0ELMA019	Undefined	32	FFEB 1130$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B19	APAA0ELMB019	Undefined	32	FFEB 1134$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C19	APAA0ELMC019	Undefined	32	FFEB 1138$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A20	APAA0ELMA020	Undefined	32	FFEB 1140$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B20	APAA0ELMB020	Undefined	32	FFEB 1144$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C20	APAA0ELMC020	Undefined	32	FFEB 1148$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A21	APAA0ELMA021	Undefined	32	FFEB 1150$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B21	APAA0ELMB021	Undefined	32	FFEB 1154$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C21	APAA0ELMC021	Undefined	32	FFEB 1158$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A22	APAA0ELMA022	Undefined	32	FFEB 1160$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B22	APAA0ELMB022	Undefined	32	FFEB 1164$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C22	APAA0ELMC022	Undefined	32	FFEB 1168$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A23	APAA0ELMA023	Undefined	32	FFEB 1170$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B23	APAA0ELMB023	Undefined	32	FFEB 1174$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C23	APAA0ELMC023	Undefined	32	FFEB 1178$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A24	APAA0ELMA024	Undefined	32	FFEB 1180$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B24	APAA0ELMB024	Undefined	32	FFEB 1184$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C24	APAA0ELMC024	Undefined	32	FFEB 1188$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A25	APAA0ELMA025	Undefined	32	FFEB 1190$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B25	APAA0ELMB025	Undefined	32	FFEB 1194$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C25	APAA0ELMC025	Undefined	32	FFEB 1198$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A26	APAA0ELMA026	Undefined	32	FFEB 11A0$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B26	APAA0ELMB026	Undefined	32	FFEB 11A4$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C26	APAA0ELMC026	Undefined	32	FFEB 11A8$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A27	APAA0ELMA027	Undefined	32	FFEB 11B0$_H$	1(1L)	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
APAA0	APAA0 Element Setting Register B27	APAA0ELMB027	Undefined	32	FFEB 11B4$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C27	APAA0ELMC027	Undefined	32	FFEB 11B8$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A28	APAA0ELMA028	Undefined	32	FFEB 11C0$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B28	APAA0ELMB028	Undefined	32	FFEB 11C4$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C28	APAA0ELMC028	Undefined	32	FFEB 11C8$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A29	APAA0ELMA029	Undefined	32	FFEB 11D0$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B29	APAA0ELMB029	Undefined	32	FFEB 11D4$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C29	APAA0ELMC029	Undefined	32	FFEB 11D8$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A30	APAA0ELMA030	Undefined	32	FFEB 11E0$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B30	APAA0ELMB030	Undefined	32	FFEB 11E4$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C30	APAA0ELMC030	Undefined	32	FFEB 11E8$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A31	APAA0ELMA031	Undefined	32	FFEB 11F0$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B31	APAA0ELMB031	Undefined	32	FFEB 11F4$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C31	APAA0ELMC031	Undefined	32	FFEB 11F8$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A32	APAA0ELMA032	Undefined	32	FFEB 1200$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B32	APAA0ELMB032	Undefined	32	FFEB 1204$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C32	APAA0ELMC032	Undefined	32	FFEB 1208$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A33	APAA0ELMA033	Undefined	32	FFEB 1210$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B33	APAA0ELMB033	Undefined	32	FFEB 1214$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C33	APAA0ELMC033	Undefined	32	FFEB 1218$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A34	APAA0ELMA034	Undefined	32	FFEB 1220$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B34	APAA0ELMB034	Undefined	32	FFEB 1224$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C34	APAA0ELMC034	Undefined	32	FFEB 1228$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A35	APAA0ELMA035	Undefined	32	FFEB 1230$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B35	APAA0ELMB035	Undefined	32	FFEB 1234$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C35	APAA0ELMC035	Undefined	32	FFEB 1238$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A36	APAA0ELMA036	Undefined	32	FFEB 1240$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B36	APAA0ELMB036	Undefined	32	FFEB 1244$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C36	APAA0ELMC036	Undefined	32	FFEB 1248$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A37	APAA0ELMA037	Undefined	32	FFEB 1250$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B37	APAA0ELMB037	Undefined	32	FFEB 1254$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C37	APAA0ELMC037	Undefined	32	FFEB 1258$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A38	APAA0ELMA038	Undefined	32	FFEB 1260$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B38	APAA0ELMB038	Undefined	32	FFEB 1264$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C38	APAA0ELMC038	Undefined	32	FFEB 1268$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A39	APAA0ELMA039	Undefined	32	FFEB 1270$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B39	APAA0ELMB039	Undefined	32	FFEB 1274$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C39	APAA0ELMC039	Undefined	32	FFEB 1278$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A40	APAA0ELMA040	Undefined	32	FFEB 1280$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B40	APAA0ELMB040	Undefined	32	FFEB 1284$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C40	APAA0ELMC040	Undefined	32	FFEB 1288$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A41	APAA0ELMA041	Undefined	32	FFEB 1290$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B41	APAA0ELMB041	Undefined	32	FFEB 1294$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C41	APAA0ELMC041	Undefined	32	FFEB 1298$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A42	APAA0ELMA042	Undefined	32	FFEB 12A0$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B42	APAA0ELMB042	Undefined	32	FFEB 12A4$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C42	APAA0ELMC042	Undefined	32	FFEB 12A8$_H$	1(1L)	32

(107/201)

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
APAA0	APAA0 Element Setting Register A43	APAA0ELMA043	Undefined	32	FFEB 12B0$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B43	APAA0ELMB043	Undefined	32	FFEB 12B4$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C43	APAA0ELMC043	Undefined	32	FFEB 12B8$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A44	APAA0ELMA044	Undefined	32	FFEB 12C0$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B44	APAA0ELMB044	Undefined	32	FFEB 12C4$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C44	APAA0ELMC044	Undefined	32	FFEB 12C8$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A45	APAA0ELMA045	Undefined	32	FFEB 12D0$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B45	APAA0ELMB045	Undefined	32	FFEB 12D4$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C45	APAA0ELMC045	Undefined	32	FFEB 12D8$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A46	APAA0ELMA046	Undefined	32	FFEB 12E0$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B46	APAA0ELMB046	Undefined	32	FFEB 12E4$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C46	APAA0ELMC046	Undefined	32	FFEB 12E8$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A47	APAA0ELMA047	Undefined	32	FFEB 12F0$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B47	APAA0ELMB047	Undefined	32	FFEB 12F4$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C47	APAA0ELMC047	Undefined	32	FFEB 12F8$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A48	APAA0ELMA048	Undefined	32	FFEB 1300$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B48	APAA0ELMB048	Undefined	32	FFEB 1304$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C48	APAA0ELMC048	Undefined	32	FFEB 1308$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A49	APAA0ELMA049	Undefined	32	FFEB 1310$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B49	APAA0ELMB049	Undefined	32	FFEB 1314$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C49	APAA0ELMC049	Undefined	32	FFEB 1318$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A50	APAA0ELMA050	Undefined	32	FFEB 1320$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B50	APAA0ELMB050	Undefined	32	FFEB 1324$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C50	APAA0ELMC050	Undefined	32	FFEB 1328$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A51	APAA0ELMA051	Undefined	32	FFEB 1330$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B51	APAA0ELMB051	Undefined	32	FFEB 1334$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C51	APAA0ELMC051	Undefined	32	FFEB 1338$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A52	APAA0ELMA052	Undefined	32	FFEB 1340$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B52	APAA0ELMB052	Undefined	32	FFEB 1344$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C52	APAA0ELMC052	Undefined	32	FFEB 1348$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A53	APAA0ELMA053	Undefined	32	FFEB 1350$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B53	APAA0ELMB053	Undefined	32	FFEB 1354$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C53	APAA0ELMC053	Undefined	32	FFEB 1358$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A54	APAA0ELMA054	Undefined	32	FFEB 1360$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B54	APAA0ELMB054	Undefined	32	FFEB 1364$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C54	APAA0ELMC054	Undefined	32	FFEB 1368$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A55	APAA0ELMA055	Undefined	32	FFEB 1370$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B55	APAA0ELMB055	Undefined	32	FFEB 1374$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C55	APAA0ELMC055	Undefined	32	FFEB 1378$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A56	APAA0ELMA056	Undefined	32	FFEB 1380$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B56	APAA0ELMB056	Undefined	32	FFEB 1384$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C56	APAA0ELMC056	Undefined	32	FFEB 1388$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A57	APAA0ELMA057	Undefined	32	FFEB 1390$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B57	APAA0ELMB057	Undefined	32	FFEB 1394$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C57	APAA0ELMC057	Undefined	32	FFEB 1398$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A58	APAA0ELMA058	Undefined	32	FFEB 13A0$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B58	APAA0ELMB058	Undefined	32	FFEB 13A4$_H$	1(1L)	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
APAA0	APAA0 Element Setting Register C58	APAA0ELMC058	Undefined	32	FFEB 13A8$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A59	APAA0ELMA059	Undefined	32	FFEB 13B0$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B59	APAA0ELMB059	Undefined	32	FFEB 13B4$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C59	APAA0ELMC059	Undefined	32	FFEB 13B8$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A60	APAA0ELMA060	Undefined	32	FFEB 13C0$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B60	APAA0ELMB060	Undefined	32	FFEB 13C4$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C60	APAA0ELMC060	Undefined	32	FFEB 13C8$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A61	APAA0ELMA061	Undefined	32	FFEB 13D0$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B61	APAA0ELMB061	Undefined	32	FFEB 13D4$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C61	APAA0ELMC061	Undefined	32	FFEB 13D8$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A62	APAA0ELMA062	Undefined	32	FFEB 13E0$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B62	APAA0ELMB062	Undefined	32	FFEB 13E4$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C62	APAA0ELMC062	Undefined	32	FFEB 13E8$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A63	APAA0ELMA063	Undefined	32	FFEB 13F0$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B63	APAA0ELMB063	Undefined	32	FFEB 13F4$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C63	APAA0ELMC063	Undefined	32	FFEB 13F8$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A64	APAA0ELMA064	Undefined	32	FFEB 1400$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B64	APAA0ELMB064	Undefined	32	FFEB 1404$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C64	APAA0ELMC064	Undefined	32	FFEB 1408$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A65	APAA0ELMA065	Undefined	32	FFEB 1410$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B65	APAA0ELMB065	Undefined	32	FFEB 1414$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C65	APAA0ELMC065	Undefined	32	FFEB 1418$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A66	APAA0ELMA066	Undefined	32	FFEB 1420$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B66	APAA0ELMB066	Undefined	32	FFEB 1424$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C66	APAA0ELMC066	Undefined	32	FFEB 1428$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A67	APAA0ELMA067	Undefined	32	FFEB 1430$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B67	APAA0ELMB067	Undefined	32	FFEB 1434$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C67	APAA0ELMC067	Undefined	32	FFEB 1438$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A68	APAA0ELMA068	Undefined	32	FFEB 1440$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B68	APAA0ELMB068	Undefined	32	FFEB 1444$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C68	APAA0ELMC068	Undefined	32	FFEB 1448$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A69	APAA0ELMA069	Undefined	32	FFEB 1450$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B69	APAA0ELMB069	Undefined	32	FFEB 1454$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C69	APAA0ELMC069	Undefined	32	FFEB 1458$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A70	APAA0ELMA070	Undefined	32	FFEB 1460$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B70	APAA0ELMB070	Undefined	32	FFEB 1464$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C70	APAA0ELMC070	Undefined	32	FFEB 1468$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A71	APAA0ELMA071	Undefined	32	FFEB 1470$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B71	APAA0ELMB071	Undefined	32	FFEB 1474$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C71	APAA0ELMC071	Undefined	32	FFEB 1478$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A72	APAA0ELMA072	Undefined	32	FFEB 1480$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B72	APAA0ELMB072	Undefined	32	FFEB 1484$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C72	APAA0ELMC072	Undefined	32	FFEB 1488$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A73	APAA0ELMA073	Undefined	32	FFEB 1490$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B73	APAA0ELMB073	Undefined	32	FFEB 1494$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C73	APAA0ELMC073	Undefined	32	FFEB 1498$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A74	APAA0ELMA074	Undefined	32	FFEB 14A0$_H$	1(1L)	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
APAA0	APAA0 Element Setting Register B74	APAA0ELMB074	Undefined	32	FFEB 14A4$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C74	APAA0ELMC074	Undefined	32	FFEB 14A8$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A75	APAA0ELMA075	Undefined	32	FFEB 14B0$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B75	APAA0ELMB075	Undefined	32	FFEB 14B4$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C75	APAA0ELMC075	Undefined	32	FFEB 14B8$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A76	APAA0ELMA076	Undefined	32	FFEB 14C0$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B76	APAA0ELMB076	Undefined	32	FFEB 14C4$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C76	APAA0ELMC076	Undefined	32	FFEB 14C8$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A77	APAA0ELMA077	Undefined	32	FFEB 14D0$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B77	APAA0ELMB077	Undefined	32	FFEB 14D4$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C77	APAA0ELMC077	Undefined	32	FFEB 14D8$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A78	APAA0ELMA078	Undefined	32	FFEB 14E0$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B78	APAA0ELMB078	Undefined	32	FFEB 14E4$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C78	APAA0ELMC078	Undefined	32	FFEB 14E8$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A79	APAA0ELMA079	Undefined	32	FFEB 14F0$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B79	APAA0ELMB079	Undefined	32	FFEB 14F4$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C79	APAA0ELMC079	Undefined	32	FFEB 14F8$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A80	APAA0ELMA080	Undefined	32	FFEB 1500$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B80	APAA0ELMB080	Undefined	32	FFEB 1504$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C80	APAA0ELMC080	Undefined	32	FFEB 1508$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A81	APAA0ELMA081	Undefined	32	FFEB 1510$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B81	APAA0ELMB081	Undefined	32	FFEB 1514$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C81	APAA0ELMC081	Undefined	32	FFEB 1518$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A82	APAA0ELMA082	Undefined	32	FFEB 1520$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B82	APAA0ELMB082	Undefined	32	FFEB 1524$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C82	APAA0ELMC082	Undefined	32	FFEB 1528$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A83	APAA0ELMA083	Undefined	32	FFEB 1530$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B83	APAA0ELMB083	Undefined	32	FFEB 1534$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C83	APAA0ELMC083	Undefined	32	FFEB 1538$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A84	APAA0ELMA084	Undefined	32	FFEB 1540$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B84	APAA0ELMB084	Undefined	32	FFEB 1544$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C84	APAA0ELMC084	Undefined	32	FFEB 1548$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A85	APAA0ELMA085	Undefined	32	FFEB 1550$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B85	APAA0ELMB085	Undefined	32	FFEB 1554$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C85	APAA0ELMC085	Undefined	32	FFEB 1558$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A86	APAA0ELMA086	Undefined	32	FFEB 1560$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B86	APAA0ELMB086	Undefined	32	FFEB 1564$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C86	APAA0ELMC086	Undefined	32	FFEB 1568$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A87	APAA0ELMA087	Undefined	32	FFEB 1570$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B87	APAA0ELMB087	Undefined	32	FFEB 1574$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C87	APAA0ELMC087	Undefined	32	FFEB 1578$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A88	APAA0ELMA088	Undefined	32	FFEB 1580$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B88	APAA0ELMB088	Undefined	32	FFEB 1584$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C88	APAA0ELMC088	Undefined	32	FFEB 1588$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A89	APAA0ELMA089	Undefined	32	FFEB 1590$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B89	APAA0ELMB089	Undefined	32	FFEB 1594$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C89	APAA0ELMC089	Undefined	32	FFEB 1598$_H$	1(1L)	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
APAA0	APAA0 Element Setting Register A90	APAA0ELMA090	Undefined	32	FFEB 15A0$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B90	APAA0ELMB090	Undefined	32	FFEB 15A4$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C90	APAA0ELMC090	Undefined	32	FFEB 15A8$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A91	APAA0ELMA091	Undefined	32	FFEB 15B0$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B91	APAA0ELMB091	Undefined	32	FFEB 15B4$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C91	APAA0ELMC091	Undefined	32	FFEB 15B8$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A92	APAA0ELMA092	Undefined	32	FFEB 15C0$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B92	APAA0ELMB092	Undefined	32	FFEB 15C4$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C92	APAA0ELMC092	Undefined	32	FFEB 15C8$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A93	APAA0ELMA093	Undefined	32	FFEB 15D0$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B93	APAA0ELMB093	Undefined	32	FFEB 15D4$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C93	APAA0ELMC093	Undefined	32	FFEB 15D8$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A94	APAA0ELMA094	Undefined	32	FFEB 15E0$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B94	APAA0ELMB094	Undefined	32	FFEB 15E4$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C94	APAA0ELMC094	Undefined	32	FFEB 15E8$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A95	APAA0ELMA095	Undefined	32	FFEB 15F0$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B95	APAA0ELMB095	Undefined	32	FFEB 15F4$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C95	APAA0ELMC095	Undefined	32	FFEB 15F8$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A96	APAA0ELMA096	Undefined	32	FFEB 1600$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B96	APAA0ELMB096	Undefined	32	FFEB 1604$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C96	APAA0ELMC096	Undefined	32	FFEB 1608$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A97	APAA0ELMA097	Undefined	32	FFEB 1610$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B97	APAA0ELMB097	Undefined	32	FFEB 1614$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C97	APAA0ELMC097	Undefined	32	FFEB 1618$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A98	APAA0ELMA098	Undefined	32	FFEB 1620$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B98	APAA0ELMB098	Undefined	32	FFEB 1624$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C98	APAA0ELMC098	Undefined	32	FFEB 1628$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A99	APAA0ELMA099	Undefined	32	FFEB 1630$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B99	APAA0ELMB099	Undefined	32	FFEB 1634$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C99	APAA0ELMC099	Undefined	32	FFEB 1638$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A100	APAA0ELMA100	Undefined	32	FFEB 1640$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B100	APAA0ELMB100	Undefined	32	FFEB 1644$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C100	APAA0ELMC100	Undefined	32	FFEB 1648$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A101	APAA0ELMA101	Undefined	32	FFEB 1650$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B101	APAA0ELMB101	Undefined	32	FFEB 1654$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C101	APAA0ELMC101	Undefined	32	FFEB 1658$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A102	APAA0ELMA102	Undefined	32	FFEB 1660$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B102	APAA0ELMB102	Undefined	32	FFEB 1664$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C102	APAA0ELMC102	Undefined	32	FFEB 1668$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A103	APAA0ELMA103	Undefined	32	FFEB 1670$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B103	APAA0ELMB103	Undefined	32	FFEB 1674$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C103	APAA0ELMC103	Undefined	32	FFEB 1678$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A104	APAA0ELMA104	Undefined	32	FFEB 1680$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B104	APAA0ELMB104	Undefined	32	FFEB 1684$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C104	APAA0ELMC104	Undefined	32	FFEB 1688$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A105	APAA0ELMA105	Undefined	32	FFEB 1690$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B105	APAA0ELMB105	Undefined	32	FFEB 1694$_H$	1(1L)	32

(111/201)

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
APAA0	APAA0 Element Setting Register C105	APAA0ELMC105	Undefined	32	FFEB 1698$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A106	APAA0ELMA106	Undefined	32	FFEB 16A0$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B106	APAA0ELMB106	Undefined	32	FFEB 16A4$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C106	APAA0ELMC106	Undefined	32	FFEB 16A8$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A107	APAA0ELMA107	Undefined	32	FFEB 16B0$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B107	APAA0ELMB107	Undefined	32	FFEB 16B4$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C107	APAA0ELMC107	Undefined	32	FFEB 16B8$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A108	APAA0ELMA108	Undefined	32	FFEB 16C0$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B108	APAA0ELMB108	Undefined	32	FFEB 16C4$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C108	APAA0ELMC108	Undefined	32	FFEB 16C8$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A109	APAA0ELMA109	Undefined	32	FFEB 16D0$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B109	APAA0ELMB109	Undefined	32	FFEB 16D4$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C109	APAA0ELMC109	Undefined	32	FFEB 16D8$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A110	APAA0ELMA110	Undefined	32	FFEB 16E0$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B110	APAA0ELMB110	Undefined	32	FFEB 16E4$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C110	APAA0ELMC110	Undefined	32	FFEB 16E8$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A111	APAA0ELMA111	Undefined	32	FFEB 16F0$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B111	APAA0ELMB111	Undefined	32	FFEB 16F4$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C111	APAA0ELMC111	Undefined	32	FFEB 16F8$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A112	APAA0ELMA112	Undefined	32	FFEB 1700$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B112	APAA0ELMB112	Undefined	32	FFEB 1704$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C112	APAA0ELMC112	Undefined	32	FFEB 1708$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A113	APAA0ELMA113	Undefined	32	FFEB 1710$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B113	APAA0ELMB113	Undefined	32	FFEB 1714$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C113	APAA0ELMC113	Undefined	32	FFEB 1718$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A114	APAA0ELMA114	Undefined	32	FFEB 1720$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B114	APAA0ELMB114	Undefined	32	FFEB 1724$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C114	APAA0ELMC114	Undefined	32	FFEB 1728$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A115	APAA0ELMA115	Undefined	32	FFEB 1730$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B115	APAA0ELMB115	Undefined	32	FFEB 1734$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C115	APAA0ELMC115	Undefined	32	FFEB 1738$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A116	APAA0ELMA116	Undefined	32	FFEB 1740$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B116	APAA0ELMB116	Undefined	32	FFEB 1744$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C116	APAA0ELMC116	Undefined	32	FFEB 1748$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A117	APAA0ELMA117	Undefined	32	FFEB 1750$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B117	APAA0ELMB117	Undefined	32	FFEB 1754$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C117	APAA0ELMC117	Undefined	32	FFEB 1758$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A118	APAA0ELMA118	Undefined	32	FFEB 1760$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B118	APAA0ELMB118	Undefined	32	FFEB 1764$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C118	APAA0ELMC118	Undefined	32	FFEB 1768$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A119	APAA0ELMA119	Undefined	32	FFEB 1770$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B119	APAA0ELMB119	Undefined	32	FFEB 1774$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C119	APAA0ELMC119	Undefined	32	FFEB 1778$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A120	APAA0ELMA120	Undefined	32	FFEB 1780$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B120	APAA0ELMB120	Undefined	32	FFEB 1784$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C120	APAA0ELMC120	Undefined	32	FFEB 1788v	1(1L)	32
APAA0	APAA0 Element Setting Register A121	APAA0ELMA121	Undefined	32	FFEB 1790$_H$	1(1L)	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
APAA0	APAA0 Element Setting Register B121	APAA0ELMB121	Undefined	32	FFEB 1794$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C121	APAA0ELMC121	Undefined	32	FFEB 1798$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A122	APAA0ELMA122	Undefined	32	FFEB 17A0$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B122	APAA0ELMB122	Undefined	32	FFEB 17A4$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C122	APAA0ELMC122	Undefined	32	FFEB 17A8$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A123	APAA0ELMA123	Undefined	32	FFEB 17B0$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B123	APAA0ELMB123	Undefined	32	FFEB 17B4$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C123	APAA0ELMC123	Undefined	32	FFEB 17B8$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A124	APAA0ELMA124	Undefined	32	FFEB 17C0$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B124	APAA0ELMB124	Undefined	32	FFEB 17C4$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C124	APAA0ELMC124	Undefined	32	FFEB 17C8$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A125	APAA0ELMA125	Undefined	32	FFEB 17D0$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B125	APAA0ELMB125	Undefined	32	FFEB 17D4$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C125	APAA0ELMC125	Undefined	32	FFEB 17D8$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A126	APAA0ELMA126	Undefined	32	FFEB 17E0$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B126	APAA0ELMB126	Undefined	32	FFEB 17E4$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C126	APAA0ELMC126	Undefined	32	FFEB 17E8$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register A127	APAA0ELMA127	Undefined	32	FFEB 17F0$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register B127	APAA0ELMB127	Undefined	32	FFEB 17F4$_H$	1(1L)	32
APAA0	APAA0 Element Setting Register C127	APAA0ELMC127	Undefined	32	FFEB 17F8$_H$	1(1L)	32
OSTM0	OSTM Comparison Register	OSTM0CMP	0000 0000$_H$	32	FFEC 0000$_H$	1(1L)	32
OSTM0	OSTM Counter Register	OSTM0CNT	FFFF FFFF$_H$	32	FFEC 0004$_H$	1(1L)	32
OSTM0	OSTM Output Register	OSTM0TO	00$_H$	8	FFEC 0008$_H$	1(1L)	8
OSTM0	OSTM Output Enable Register	OSTM0TOE	00$_H$	8	FFEC 000C$_H$	1(1L)	8
OSTM0	OSTM Count Enable Status Register	OSTM0TE	00$_H$	8	FFEC 0010$_H$	1(1L)	8
OSTM0	OSTM Count Start Trigger Register	OSTM0TS	00$_H$	8	FFEC 0014$_H$	1(1L)	8
OSTM0	OSTM Count Stop Trigger Register	OSTM0TT	00$_H$	8	FFEC 0018$_H$	1(1L)	8
OSTM0	OSTM Control Register	OSTM0CTL	00$_H$	8	FFEC 0020$_H$	1(1L)	8
OSTM1	OSTM Comparison Register	OSTM1CMP	0000 0000$_H$	32	FFEC 1000$_H$	1(1L)	32
OSTM1	OSTM Counter Register	OSTM1CNT	FFFF FFFF$_H$	32	FFEC 1004$_H$	1(1L)	32
OSTM1	OSTM Output Register	OSTM1TO	00$_H$	8	FFEC 1008$_H$	1(1L)	8
OSTM1	OSTM Output Enable Register	OSTM1TOE	00$_H$	8	FFEC 100C$_H$	1(1L)	8
OSTM1	OSTM Count Enable Status Register	OSTM1TE	00$_H$	8	FFEC 1010$_H$	1(1L)	8
OSTM1	OSTM Count Start Trigger Register	OSTM1TS	00$_H$	8	FFEC 1014$_H$	1(1L)	8
OSTM1	OSTM Count Stop Trigger Register	OSTM1TT	00$_H$	8	FFEC 1018$_H$	1(1L)	8
OSTM1	OSTM Control Register	OSTM1CTL	00$_H$	8	FFEC 1020$_H$	1(1L)	8
OSTM2	OSTM Comparison Register	OSTM2CMP	0000 0000$_H$	32	FFEC 2000$_H$	1(1L)	32
OSTM2	OSTM Counter Register	OSTM2CNT	FFFF FFFF$_H$	32	FFEC 2004$_H$	1(1L)	32
OSTM2	OSTM Output Register	OSTM2TO	00$_H$	8	FFEC 2008$_H$	1(1L)	8
OSTM2	OSTM Output Enable Register	OSTM2TOE	00$_H$	8	FFEC 200C$_H$	1(1L)	8
OSTM2	OSTM Count Enable Status Register	OSTM2TE	00$_H$	8	FFEC 2010$_H$	1(1L)	8
OSTM2	OSTM Count Start Trigger Register	OSTM2TS	00$_H$	8	FFEC 2014$_H$	1(1L)	8
OSTM2	OSTM Count Stop Trigger Register	OSTM2TT	00$_H$	8	FFEC 2018$_H$	1(1L)	8
OSTM2	OSTM Control Register	OSTM2CTL	00$_H$	8	FFEC 2020$_H$	1(1L)	8
WDTA0	WDTA Enable Register	WDTA0WDTE	2C$_H$	8	FFED 0000$_H$	1(1L)	8
WDTA0	WDTA Mode Register	WDTA0MD	7F$_H$	8	FFED 000C$_H$	1(1L)	8
WDTA1	WDTA Enable Register	WDTA1WDTE	2C$_H$	8	FFED 1000$_H$	1(1L)	8

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
WDTA1	WDTA Mode Register	WDTA1MD	$7F_H$	8	FFED $100C_H$	1(1L)	8
APGA0	APGA0 Register Control Register 0	APGA0CNTR0	00_H	8	FFED $C000_H$	1(1L)	8
APGA0	APGA0 Register Control Register 1	APGA0CNTR1	00_H	8	FFED $C004_H$	1(1L)	8
APGA0	APGA0 Register Control Register 3	APGA0CNTR3	FF_H	8	FFED $C005_H$	1(1L)	8
APGA0	APGA0 Register Control Register 2	APGA0CNTR2	0000 0000_H	32	FFED $C008_H$	1(1L)	32
RHSBG0	Microsecond Bus Control Register H	RHSBG0CR0H	0000 0000_H	32	FFED $D000_H$	1(1L)	8, 16, 32
RHSBG0	Microsecond Bus Control Register L	RHSBG0CR0L	0000 0000_H	32	FFED $D004_H$	1(1L)	8, 16, 32
RHSBG0	Microsecond Bus Control Register H	RHSBG0CR1H	0000 0000_H	32	FFED $D008_H$	1(1L)	8, 16, 32
RHSBG0	Microsecond Bus Control Register L	RHSBG0CR1L	0000 0000_H	32	FFED $D00C_H$	1(1L)	8, 16, 32
RHSBG0	Microsecond Bus Control Register H	RHSBG0CR2H	0000 0000_H	32	FFED $D010_H$	1(1L)	8, 16, 32
RHSBG0	Microsecond Bus Control Register L	RHSBG0CR2L	0000 0000_H	32	FFED $D014_H$	1(1L)	8, 16, 32
RHSBG0	Microsecond Bus Control Register H	RHSBG0CR3H	0000 0000_H	32	FFED $D018_H$	1(1L)	8, 16, 32
RHSBG0	Microsecond Bus Control Register L	RHSBG0CR3L	0000 0000_H	32	FFED $D01C_H$	1(1L)	8, 16, 32
RHSBG1	Microsecond Bus Control Register H	RHSBG1CR0H	0000 0000_H	32	FFED $D800_H$	1(1L)	8, 16, 32
RHSBG1	Microsecond Bus Control Register L	RHSBG1CR0L	0000 0000_H	32	FFED $D804_H$	1(1L)	8, 16, 32
RHSBG1	Microsecond Bus Control Register H	RHSBG1CR1H	0000 0000_H	32	FFED $D808_H$	1(1L)	8, 16, 32
RHSBG1	Microsecond Bus Control Register L	RHSBG1CR1L	0000 0000_H	32	FFED $D80C_H$	1(1L)	8, 16, 32
RHSBG1	Microsecond Bus Control Register H	RHSBG1CR2H	0000 0000_H	32	FFED $D810_H$	1(1L)	8, 16, 32
RHSBG1	Microsecond Bus Control Register L	RHSBG1CR2L	0000 0000_H	32	FFED $D814_H$	1(1L)	8, 16, 32
RHSBG1	Microsecond Bus Control Register H	RHSBG1CR3H	0000 0000_H	32	FFED $D818_H$	1(1L)	8, 16, 32
RHSBG1	Microsecond Bus Control Register L	RHSBG1CR3L	0000 0000_H	32	FFED $D81C_H$	1(1L)	8, 16, 32
RHSB0	Global Setting Register	RHSB0GC	0000 0000_H	32	FFEE 0000_H	3	8, 16, 32
RHSB0	Module Status Register	RHSB0MSR	0001 0000_H	32	FFEE 0008_H	3	8, 16, 32
RHSB0	Down Stream Setting Register	RHSB0DCR	0011 $FF00_H$	32	FFEE 0010_H	3	16, 32
RHSB0	Data Element Setting Register	RHSB0DEC	0000 0000_H	32	FFEE 0014_H	3	8, 16, 32
RHSB0	Slave Device Setting Register 0	RHSB0SDC0	0000 0000_H	32	FFEE 0018_H	3	16, 32
RHSB0	Data Element Bit Assignment Register 0	RHSB0DEBA0	0000 0000_H	32	FFEE 0020_H	3	16, 32
RHSB0	Data Element Bit Assignment Register 1	RHSB0DEBA1	0000 0000_H	32	FFEE 0024_H	3	16, 32
RHSB0	Data Element Bit Assignment Register 2	RHSB0DEBA2	0000 0000_H	32	FFEE 0028_H	3	16, 32
RHSB0	Data Element Bit Assignment Register 3	RHSB0DEBA3	0000 0000_H	32	FFEE $002C_H$	3	16, 32
RHSB0	Emergency Bit Enable Register 0	RHSB0EBE0	0000 0000_H	32	FFEE 0030_H	3	16, 32
RHSB0	Emergency Bit Enable Register 1	RHSB0EBE1	0000 0000_H	32	FFEE 0034_H	3	16, 32
RHSB0	Down Stream Transmission Control Register	RHSB0DTC	0000 0000_H	32	FFEE 0038_H	3	8, 16, 32
RHSB0	Down Stream Command Data Register	RHSB0DCD	0000 0000_H	32	FFEE $003C_H$	3	16, 32
RHSB0	Down Stream Data Register 0	RHSB0DDR0	0000 0000_H	32	FFEE 0040_H	3	16, 32
RHSB0	Down Stream Data Register 1	RHSB0DDR1	0000 0000_H	32	FFEE 0044_H	3	16, 32
RHSB0	Down Stream Emergency Data Register 0	RHSB0DED0	0000 0000_H	32	FFEE 0048_H	3	16, 32
RHSB0	Down Stream Emergency Data Register 1	RHSB0DED1	0000 0000_H	32	FFEE $004C_H$	3	16, 32
RHSB0	Up Stream Setting Register	RHSB0UCR	0000 1800_H	32	FFEE 0050_H	3	8, 16, 32
RHSB0	Up Stream Channel Setting Register	RHSB0UCC	0F0F $0F0F_H$	32	FFEE 0054_H	3	8, 16, 32
RHSB0	Up Stream Channel Selection Register	RHSB0UCS	0000 0000_H	32	FFEE 0058_H	3	8, 16, 32
RHSB0	Up Stream Data Read Register	RHSB0UDR	0000 0000_H	32	FFEE $005C_H$	3	8, 16, 32
RHSB0	Up Stream Data Register 0	RHSB0UD0	0000 0000_H	32	FFEE 0060_H	3	8, 16, 32
RHSB0	Up Stream Data Register 1	RHSB0UD1	0000 0000_H	32	FFEE 0064_H	3	8, 16, 32
RHSB0	Up Stream Status Summary Register	RHSB0USS	0000 0000_H	32	FFEE 0070_H	3	8, 16, 32
RHSB0	Interrupt Control Register	RHSB0IC	0000 0000_H	32	FFEE 0074_H	3	8, 16, 32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
RHSB0	Interrupt Status Register	RHSB0IS	0000 0000$_H$	32	FFEE 0078$_H$	3	8, 16, 32
RHSB0	Down Stream Setting Register for Cycle 1	RHSB0DCR1	0000 0000$_H$	32	FFEE 007C$_H$	3	8, 16, 32
RHSB1	Global Setting Register	RHSB1GC	0000 0000$_H$	32	FFEE 1000$_H$	3	8, 16, 32
RHSB1	Module Status Register	RHSB1MSR	0001 0000$_H$	32	FFEE 1008$_H$	3	8, 16, 32
RHSB1	Down Stream Setting Register	RHSB1DCR	0011 FF00$_H$	32	FFEE 1010$_H$	3	16, 32
RHSB1	Data Element Setting Register	RHSB1DEC	0000 0000$_H$	32	FFEE 1014$_H$	3	8, 16, 32
RHSB1	Slave Device Setting Register 0	RHSB1SDC0	0000 0000$_H$	32	FFEE 1018$_H$	3	16, 32
RHSB1	Data Element Bit Assignment Register 0	RHSB1DEBA0	0000 0000$_H$	32	FFEE 1020$_H$	3	16, 32
RHSB1	Data Element Bit Assignment Register 1	RHSB1DEBA1	0000 0000$_H$	32	FFEE 1024$_H$	3	16, 32
RHSB1	Data Element Bit Assignment Register 2	RHSB1DEBA2	0000 0000$_H$	32	FFEE 1028$_H$	3	16, 32
RHSB1	Data Element Bit Assignment Register 3	RHSB1DEBA3	0000 0000$_H$	32	FFEE 102C$_H$	3	16, 32
RHSB1	Emergency Bit Enable Register 0	RHSB1EBE0	0000 0000$_H$	32	FFEE 1030$_H$	3	16, 32
RHSB1	Emergency Bit Enable Register 1	RHSB1EBE1	0000 0000$_H$	32	FFEE 1034$_H$	3	16, 32
RHSB1	Down Stream Transmission Control Register	RHSB1DTC	0000 0000$_H$	32	FFEE 1038$_H$	3	8, 16, 32
RHSB1	Down Stream Command Data Register	RHSB1DCD	0000 0000$_H$	32	FFEE 103C$_H$	3	16, 32
RHSB1	Down Stream Data Register 0	RHSB1DDR0	0000 0000$_H$	32	FFEE 1040$_H$	3	16, 32
RHSB1	Down Stream Data Register 1	RHSB1DDR1	0000 0000$_H$	32	FFEE 1044$_H$	3	16, 32
RHSB1	Down Stream Emergency Data Register 0	RHSB1DED0	0000 0000$_H$	32	FFEE 1048$_H$	3	16, 32
RHSB1	Down Stream Emergency Data Register 1	RHSB1DED1	0000 0000$_H$	32	FFEE 104C$_H$	3	16, 32
RHSB1	Up Stream Setting Register	RHSB1UCR	0000 1800$_H$	32	FFEE 1050$_H$	3	8, 16, 32
RHSB1	Up Stream Channel Setting Register	RHSB1UCC	0F0F 0F0F$_H$	32	FFEE 1054$_H$	3	8, 16, 32
RHSB1	Up Stream Channel Selection Register	RHSB1UCS	0000 0000$_H$	32	FFEE 1058$_H$	3	8, 16, 32
RHSB1	Up Stream Data Read Register	RHSB1UDR	0000 0000$_H$	32	FFEE 105C$_H$	3	8, 16, 32
RHSB1	Up Stream Data Register 0	RHSB1UD0	0000 0000$_H$	32	FFEE 1060$_H$	3	8, 16, 32
RHSB1	Up Stream Data Register 1	RHSB1UD1	0000 0000$_H$	32	FFEE 1064$_H$	3	8, 16, 32
RHSB1	Up Stream Status Summary Register	RHSB1USS	0000 0000$_H$	32	FFEE 1070$_H$	3	8, 16, 32
RHSB1	Interrupt Control Register	RHSB1IC	0000 0000$_H$	32	FFEE 1074$_H$	3	8, 16, 32
RHSB1	Interrupt Status Register	RHSB1IS	0000 0000$_H$	32	FFEE 1078$_H$	3	8, 16, 32
RHSB1	Down Stream Setting Register for Cycle 1	RHSB1DCR1	0000 0000$_H$	32	FFEE 107C$_H$	3	8, 16, 32
ADCB0	Virtual Channel Register 0	ADCB0VCR00	0000$_H$	16	FFF2 0000$_H$	3	8, 16
ADCB0	Virtual Channel Register 1	ADCB0VCR01	0000$_H$	16	FFF2 0004$_H$	3	8, 16
ADCB0	Virtual Channel Register 2	ADCB0VCR02	0000$_H$	16	FFF2 0008$_H$	3	8, 16
ADCB0	Virtual Channel Register 3	ADCB0VCR03	0000$_H$	16	FFF2 000C$_H$	3	8, 16
ADCB0	Virtual Channel Register 4	ADCB0VCR04	0000$_H$	16	FFF2 0010$_H$	3	8, 16
ADCB0	Virtual Channel Register 5	ADCB0VCR05	0000$_H$	16	FFF2 0014$_H$	3	8, 16
ADCB0	Virtual Channel Register 6	ADCB0VCR06	0000$_H$	16	FFF2 0018$_H$	3	8, 16
ADCB0	Virtual Channel Register 7	ADCB0VCR07	0000$_H$	16	FFF2 001C$_H$	3	8, 16
ADCB0	Virtual Channel Register 8	ADCB0VCR08	0000$_H$	16	FFF2 0020$_H$	3	8, 16
ADCB0	Virtual Channel Register 9	ADCB0VCR09	0000$_H$	16	FFF2 0024$_H$	3	8, 16
ADCB0	Virtual Channel Register 10	ADCB0VCR10	0000$_H$	16	FFF2 0028$_H$	3	8, 16
ADCB0	Virtual Channel Register 11	ADCB0VCR11	0000$_H$	16	FFF2 002C$_H$	3	8, 16
ADCB0	Virtual Channel Register 12	ADCB0VCR12	0000$_H$	16	FFF2 0030$_H$	3	8, 16
ADCB0	Virtual Channel Register 13	ADCB0VCR13	0000$_H$	16	FFF2 0034$_H$	3	8, 16
ADCB0	Virtual Channel Register 14	ADCB0VCR14	0000$_H$	16	FFF2 0038$_H$	3	8, 16
ADCB0	Virtual Channel Register 15	ADCB0VCR15	0000$_H$	16	FFF2 003C$_H$	3	8, 16
ADCB0	Virtual Channel Register 16	ADCB0VCR16	0000$_H$	16	FFF2 0040$_H$	3	8, 16

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
ADCB0	Virtual Channel Register 17	ADCB0VCR17	0000_H	16	FFF2 0044_H	3	8, 16
ADCB0	Virtual Channel Register 18	ADCB0VCR18	0000_H	16	FFF2 0048_H	3	8, 16
ADCB0	Virtual Channel Register 19	ADCB0VCR19	0000_H	16	FFF2 $004C_H$	3	8, 16
ADCB0	Virtual Channel Register 20	ADCB0VCR20	0000_H	16	FFF2 0050_H	3	8, 16
ADCB0	Virtual Channel Register 21	ADCB0VCR21	0000_H	16	FFF2 0054_H	3	8, 16
ADCB0	Virtual Channel Register 22	ADCB0VCR22	0000_H	16	FFF2 0058_H	3	8, 16
ADCB0	Virtual Channel Register 23	ADCB0VCR23	0000_H	16	FFF2 $005C_H$	3	8, 16
ADCB0	Virtual Channel Register 24	ADCB0VCR24	0000_H	16	FFF2 0060_H	3	8, 16
ADCB0	Virtual Channel Register 25	ADCB0VCR25	0000_H	16	FFF2 0064_H	3	8, 16
ADCB0	Virtual Channel Register 26	ADCB0VCR26	0000_H	16	FFF2 0068_H	3	8, 16
ADCB0	Virtual Channel Register 27	ADCB0VCR27	0000_H	16	FFF2 $006C_H$	3	8, 16
ADCB0	Virtual Channel Register 28	ADCB0VCR28	0000_H	16	FFF2 0070_H	3	8, 16
ADCB0	Virtual Channel Register 29	ADCB0VCR29	0000_H	16	FFF2 0074_H	3	8, 16
ADCB0	Virtual Channel Register 30	ADCB0VCR30	0000_H	16	FFF2 0078_H	3	8, 16
ADCB0	Virtual Channel Register 31	ADCB0VCR31	0000_H	16	FFF2 $007C_H$	3	8, 16
ADCB0	Virtual Channel Register 32	ADCB0VCR32	0000_H	16	FFF2 0080_H	3	8, 16
ADCB0	Virtual Channel Register 33	ADCB0VCR33	0000_H	16	FFF2 0084_H	3	8, 16
ADCB0	Virtual Channel Register 34	ADCB0VCR34	0000_H	16	FFF2 0088_H	3	8, 16
ADCB0	Virtual Channel Register 35	ADCB0VCR35	0000_H	16	FFF2 $008C_H$	3	8, 16
ADCB0	Virtual Channel Register 36	ADCB0VCR36	0000_H	16	FFF2 0090_H	3	8, 16
ADCB0	Virtual Channel Register 37	ADCB0VCR37	0000_H	16	FFF2 0094_H	3	8, 16
ADCB0	Virtual Channel Register 38	ADCB0VCR38	0000_H	16	FFF2 0098_H	3	8, 16
ADCB0	Virtual Channel Register 39	ADCB0VCR39	0000_H	16	FFF2 $009C_H$	3	8, 16
ADCB0	Data Register 0	ADCB0DR00	0000_H	16	FFF2 0100_H	3	16, 32
ADCB0	Data Register 1	ADCB0DR01	0000_H	16	FFF2 0102_H	3	16
ADCB0	Data Register 2	ADCB0DR02	0000_H	16	FFF2 0104_H	3	16, 32
ADCB0	Data Register 3	ADCB0DR03	0000_H	16	FFF2 0106_H	3	16
ADCB0	Data Register 4	ADCB0DR04	0000_H	16	FFF2 0108_H	3	16, 32
ADCB0	Data Register 5	ADCB0DR05	0000_H	16	FFF2 $010A_H$	3	16
ADCB0	Data Register 6	ADCB0DR06	0000_H	16	FFF2 $010C_H$	3	16, 32
ADCB0	Data Register 7	ADCB0DR07	0000_H	16	FFF2 $010E_H$	3	16
ADCB0	Data Register 8	ADCB0DR08	0000_H	16	FFF2 0110_H	3	16, 32
ADCB0	Data Register 9	ADCB0DR09	0000_H	16	FFF2 0112_H	3	16
ADCB0	Data Register 10	ADCB0DR10	0000_H	16	FFF2 0114_H	3	16, 32
ADCB0	Data Register 11	ADCB0DR11	0000_H	16	FFF2 0116_H	3	16
ADCB0	Data Register 12	ADCB0DR12	0000_H	16	FFF2 0118_H	3	16, 32
ADCB0	Data Register 13	ADCB0DR13	0000_H	16	FFF2 $011A_H$	3	16
ADCB0	Data Register 14	ADCB0DR14	0000_H	16	FFF2 $011C_H$	3	16, 32
ADCB0	Data Register 15	ADCB0DR15	0000_H	16	FFF2 $011E_H$	3	16
ADCB0	Data Register 16	ADCB0DR16	0000_H	16	FFF2 0120_H	3	16, 32
ADCB0	Data Register 17	ADCB0DR17	0000_H	16	FFF2 0122_H	3	16
ADCB0	Data Register 18	ADCB0DR18	0000_H	16	FFF2 0124_H	3	16, 32
ADCB0	Data Register 19	ADCB0DR19	0000_H	16	FFF2 0126_H	3	16
ADCB0	Data Register 20	ADCB0DR20	0000_H	16	FFF2 0128_H	3	16, 32
ADCB0	Data Register 21	ADCB0DR21	0000_H	16	FFF2 $012A_H$	3	16
ADCB0	Data Register 22	ADCB0DR22	0000_H	16	FFF2 $012C_H$	3	16, 32
ADCB0	Data Register 23	ADCB0DR23	0000_H	16	FFF2 $012E_H$	3	16

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
ADCB0	Data Register 24	ADCB0DR24	0000_H	16	FFF2 0130_H	3	16, 32
ADCB0	Data Register 25	ADCB0DR25	0000_H	16	FFF2 0132_H	3	16
ADCB0	Data Register 26	ADCB0DR26	0000_H	16	FFF2 0134_H	3	16, 32
ADCB0	Data Register 27	ADCB0DR27	0000_H	16	FFF2 0136_H	3	16
ADCB0	Data Register 28	ADCB0DR28	0000_H	16	FFF2 0138_H	3	16, 32
ADCB0	Data Register 29	ADCB0DR29	0000_H	16	FFF2 $013A_H$	3	16
ADCB0	Data Register 30	ADCB0DR30	0000_H	16	FFF2 $013C_H$	3	16, 32
ADCB0	Data Register 31	ADCB0DR31	0000_H	16	FFF2 $013E_H$	3	16
ADCB0	Data Register 32	ADCB0DR32	0000_H	16	FFF2 0140_H	3	16, 32
ADCB0	Data Register 33	ADCB0DR33	0000_H	16	FFF2 0142_H	3	16
ADCB0	Data Register 34	ADCB0DR34	0000_H	16	FFF2 0144_H	3	16, 32
ADCB0	Data Register 35	ADCB0DR35	0000_H	16	FFF2 0146_H	3	16
ADCB0	Data Register 36	ADCB0DR36	0000_H	16	FFF2 0148_H	3	16, 32
ADCB0	Data Register 37	ADCB0DR37	0000_H	16	FFF2 $014A_H$	3	16
ADCB0	Data Register 38	ADCB0DR38	0000_H	16	FFF2 $014C_H$	3	16, 32
ADCB0	Data Register 39	ADCB0DR39	0000_H	16	FFF2 $014E_H$	3	16
ADCB0	Data Supplementary Information Register 0	ADCB0DIR00	$0000\ 0000_H$	32	FFF2 0200_H	3	32
ADCB0	Data Supplementary Information Register 1	ADCB0DIR01	$0000\ 0000_H$	32	FFF2 0204_H	3	32
ADCB0	Data Supplementary Information Register 2	ADCB0DIR02	$0000\ 0000_H$	32	FFF2 0208_H	3	32
ADCB0	Data Supplementary Information Register 3	ADCB0DIR03	$0000\ 0000_H$	32	FFF2 $020C_H$	3	32
ADCB0	Data Supplementary Information Register 4	ADCB0DIR04	$0000\ 0000_H$	32	FFF2 0210_H	3	32
ADCB0	Data Supplementary Information Register 5	ADCB0DIR05	$0000\ 0000_H$	32	FFF2 0214_H	3	32
ADCB0	Data Supplementary Information Register 6	ADCB0DIR06	$0000\ 0000_H$	32	FFF2 0218_H	3	32
ADCB0	Data Supplementary Information Register 7	ADCB0DIR07	$0000\ 0000_H$	32	FFF2 $021C_H$	3	32
ADCB0	Data Supplementary Information Register 8	ADCB0DIR08	$0000\ 0000_H$	32	FFF2 0220_H	3	32
ADCB0	Data Supplementary Information Register 9	ADCB0DIR09	$0000\ 0000_H$	32	FFF2 0224_H	3	32
ADCB0	Data Supplementary Information Register 10	ADCB0DIR10	$0000\ 0000_H$	32	FFF2 0228_H	3	32
ADCB0	Data Supplementary Information Register 11	ADCB0DIR11	$0000\ 0000_H$	32	FFF2 $022C_H$	3	32
ADCB0	Data Supplementary Information Register 12	ADCB0DIR12	$0000\ 0000_H$	32	FFF2 0230_H	3	32
ADCB0	Data Supplementary Information Register 13	ADCB0DIR13	$0000\ 0000_H$	32	FFF2 0234_H	3	32
ADCB0	Data Supplementary Information Register 14	ADCB0DIR14	$0000\ 0000_H$	32	FFF2 0238_H	3	32
ADCB0	Data Supplementary Information Register 15	ADCB0DIR15	$0000\ 0000_H$	32	FFF2 $023C_H$	3	32
ADCB0	Data Supplementary Information Register 16	ADCB0DIR16	$0000\ 0000_H$	32	FFF2 0240_H	3	32
ADCB0	Data Supplementary Information Register 17	ADCB0DIR17	$0000\ 0000_H$	32	FFF2 0244_H	3	32
ADCB0	Data Supplementary Information Register 18	ADCB0DIR18	$0000\ 0000_H$	32	FFF2 0248_H	3	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
ADCB0	Data Supplementary Information Register 19	ADCB0DIR19	0000 0000$_H$	32	FFF2 024C$_H$	3	32
ADCB0	Data Supplementary Information Register 20	ADCB0DIR20	0000 0000$_H$	32	FFF2 0250$_H$	3	32
ADCB0	Data Supplementary Information Register 21	ADCB0DIR21	0000 0000$_H$	32	FFF2 0254$_H$	3	32
ADCB0	Data Supplementary Information Register 22	ADCB0DIR22	0000 0000$_H$	32	FFF2 0258$_H$	3	32
ADCB0	Data Supplementary Information Register 23	ADCB0DIR23	0000 0000$_H$	32	FFF2 025C$_H$	3	32
ADCB0	Data Supplementary Information Register 24	ADCB0DIR24	0000 0000$_H$	32	FFF2 0260$_H$	3	32
ADCB0	Data Supplementary Information Register 25	ADCB0DIR25	0000 0000$_H$	32	FFF2 0264$_H$	3	32
ADCB0	Data Supplementary Information Register 26	ADCB0DIR26	0000 0000$_H$	32	FFF2 0268$_H$	3	32
ADCB0	Data Supplementary Information Register 27	ADCB0DIR27	0000 0000$_H$	32	FFF2 026C$_H$	3	32
ADCB0	Data Supplementary Information Register 28	ADCB0DIR28	0000 0000$_H$	32	FFF2 0270$_H$	3	32
ADCB0	Data Supplementary Information Register 29	ADCB0DIR29	0000 0000$_H$	32	FFF2 0274$_H$	3	32
ADCB0	Data Supplementary Information Register 30	ADCB0DIR30	0000 0000$_H$	32	FFF2 0278$_H$	3	32
ADCB0	Data Supplementary Information Register 31	ADCB0DIR31	0000 0000$_H$	32	FFF2 027C$_H$	3	32
ADCB0	Data Supplementary Information Register 32	ADCB0DIR32	0000 0000$_H$	32	FFF2 0280$_H$	3	32
ADCB0	Data Supplementary Information Register 33	ADCB0DIR33	0000 0000$_H$	32	FFF2 0284$_H$	3	32
ADCB0	Data Supplementary Information Register 34	ADCB0DIR34	0000 0000$_H$	32	FFF2 0288$_H$	3	32
ADCB0	Data Supplementary Information Register 35	ADCB0DIR35	0000 0000$_H$	32	FFF2 028C$_H$	3	32
ADCB0	Data Supplementary Information Register 36	ADCB0DIR36	0000 0000$_H$	32	FFF2 0290$_H$	3	32
ADCB0	Data Supplementary Information Register 37	ADCB0DIR37	0000 0000$_H$	32	FFF2 0294$_H$	3	32
ADCB0	Data Supplementary Information Register 38	ADCB0DIR38	0000 0000$_H$	32	FFF2 0298$_H$	3	32
ADCB0	Data Supplementary Information Register 39	ADCB0DIR39	0000 0000$_H$	32	FFF2 029C$_H$	3	32
ADCB0	AD Synchronization Start Control Register	ADCB0ADSYNSTCR	00$_H$	8	FFF2 0300$_H$	3	8
ADCB0	AD Timer Synchronization Start Control Register	ADCB0ADTSYNSTCR	00$_H$	8	FFF2 0304$_H$	3	8
ADCB0	AD End Register	ADCB0ADHALTR	00$_H$	8	FFF2 0380$_H$	3	8
ADCB0	AD Control Register 1	ADCB0ADCR1	00$_H$	8	FFF2 0384$_H$	3	8
ADCB0	MPX Current Control Register	ADCB0MPXCURCR	00$_H$	8	FFF2 0388$_H$	3	8
ADCB0	MPX Current Register	ADCB0MPXCURR	0000 0000$_H$	32	FFF2 038C$_H$	3	32
ADCB0	MPX Optional Wait Register	ADCB0MPXOWR	00$_H$	8	FFF2 0390$_H$	3	8
ADCB0	MPX Command Information Register	ADCB0MPXCMDR	00$_H$	8	FFF2 0394$_H$	3	8
ADCB0	AD Control Register 2	ADCB0ADCR2	00$_H$	8	FFF2 0398$_H$	3	8
ADCB0	DFE/ASF Entry Scan Group Enable Register	ADCB0DFASENTSGER	0000$_H$	16	FFF2 039C$_H$	3	16
ADCB0	AD Change Monitor Virtual Channel Pointer	ADCB0ADENDP	00$_H$	8	FFF2 03A0$_H$	3	8

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
ADCB0	Safety Control Register	ADCB0SFTCR	00_H	8	FFF2 $03C0_H$	3	8
ADCB0	Pin Level Self-diagnostic Control Register	ADCB0TDCR	00_H	8	FFF2 $03C4_H$	3	8
ADCB0	Wiring-break Detection Control Register	ADCB0ODCR	00_H	8	FFF2 $03C8_H$	3	8
ADCB0	Upper-Limit / Lower-Limit Table Register 0	ADCB0ULLMTBR0	7FFE 0000_H	32	FFF2 $03CC_H$	3	16, 32
ADCB0	Upper-Limit / Lower-Limit Table Register 1	ADCB0ULLMTBR1	7FFE 0000_H	32	FFF2 $03D0_H$	3	16, 32
ADCB0	Upper-Limit / Lower-Limit Table Register 2	ADCB0ULLMTBR2	7FFE 0000_H	32	FFF2 $03D4_H$	3	16, 32
ADCB0	Error Clear Register	ADCB0ECR	00_H	8	FFF2 $03D8_H$	3	8
ADCB0	Upper-Limit / Lower-Limit Error Register	ADCB0ULER	00_H	8	FFF2 $03DC_H$	3	8
ADCB0	Overwrite Error Register	ADCB0OWER	00_H	8	FFF2 $03E0_H$	3	8
ADCB0	Parity Error Register	ADCB0PER	00_H	8	FFF2 $03E4_H$	3	8
ADCB0	ID Error Register	ADCB0IDER	00_H	8	FFF2 $03E8_H$	3	8
ADCB0	Scan Group 0 Start Control Register	ADCB0SGSTCR0	00_H	8	FFF2 0480_H	3	8
ADCB0	Scan Group 0 Control Register	ADCB0SGCR0	00_H	8	FFF2 0490_H	3	8
ADCB0	Scan Group 0 Start Virtual Channel Pointer	ADCB0SGVCSP0	00_H	8	FFF2 0494_H	3	8
ADCB0	Scan Group 0 End Virtual Channel Pointer	ADCB0SGVCEP0	00_H	8	FFF2 0498_H	3	8
ADCB0	Scan Group 0 Multi Cycle Register	ADCB0SGMCYCR0	00_H	8	FFF2 $049C_H$	3	8
ADCB0	Scan Group 0 Status Register	ADCB0SGSR0	00_H	8	FFF2 $04A4_H$	3	8
ADCB0	Scan Group 0 Upper-limit / Lower-limit Table Selection Register	ADCB0ULLMSR0	00_H	8	FFF2 $04B0_H$	3	8
ADCB0	Scan Group 1 Start Control Register	ADCB0SGSTCR1	00_H	8	FFF2 0500_H	3	8
ADCB0	Scan Group 1 Control Register	ADCB0SGCR1	00_H	8	FFF2 0510_H	3	8
ADCB0	Scan Group 1 Start Virtual Channel Pointer	ADCB0SGVCSP1	00_H	8	FFF2 0514_H	3	8
ADCB0	Scan Group 1 End Virtual Channel Pointer	ADCB0SGVCEP1	00_H	8	FFF2 0518_H	3	8
ADCB0	Scan Group 1 Multi Cycle Register	ADCB0SGMCYCR1	00_H	8	FFF2 $051C_H$	3	8
ADCB0	Scan Group 1 Status Register	ADCB0SGSR1	00_H	8	FFF2 0524_H	3	8
ADCB0	Scan Group 1 Upper-limit / Lower-limit Table Selection Register	ADCB0ULLMSR1	00_H	8	FFF2 0530_H	3	8
ADCB0	Scan Group 2 Start Control Register	ADCB0SGSTCR2	00_H	8	FFF2 0580_H	3	8
ADCB0	Scan Group 2 Control Register	ADCB0SGCR2	00_H	8	FFF2 0590_H	3	8
ADCB0	Scan Group 2 Start Virtual Channel Pointer	ADCB0SGVCSP2	00_H	8	FFF2 0594_H	3	8
ADCB0	Scan Group 2 End Virtual Channel Pointer	ADCB0SGVCEP2	00_H	8	FFF2 0598_H	3	8
ADCB0	Scan Group 2 Multi Cycle Register	ADCB0SGMCYCR2	00_H	8	FFF2 $059C_H$	3	8
ADCB0	Scan Group 2 Status Register	ADCB0SGSR2	00_H	8	FFF2 $05A4_H$	3	8
ADCB0	Scan Group 2 Upper-limit / Lower-limit Table Selection Register	ADCB0ULLMSR2	00_H	8	FFF2 $05B0_H$	3	8
ADCB0	Scan Group 3 Start Control Register	ADCB0SGSTCR3	00_H	8	FFF2 0600_H	3	8
ADCB0	AD Timer 3 Start Control Register	ADCB0ADTSTCR3	00_H	8	FFF2 0608_H	3	8
ADCB0	AD Timer 3 End Control Register	ADCB0ADTENDCR3	00_H	8	FFF2 $060C_H$	3	8
ADCB0	Scan Group 3 Control Register	ADCB0SGCR3	00_H	8	FFF2 0610_H	3	8
ADCB0	Scan Group 3 Start Virtual Channel Pointer	ADCB0SGVCSP3	00_H	8	FFF2 0614_H	3	8
ADCB0	Scan Group 3 End Virtual Channel Pointer	ADCB0SGVCEP3	00_H	8	FFF2 0618_H	3	8
ADCB0	Scan Group 3 Multi Cycle Register	ADCB0SGMCYCR3	00_H	8	FFF2 $061C_H$	3	8
ADCB0	Scan Group 3 Status Register	ADCB0SGSR3	00_H	8	FFF2 0624_H	3	8

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
ADCB0	AD Timer Initial Phase Register 3	ADCB0ADTIPR3	0000 0000_H	32	FFF2 0628_H	3	32
ADCB0	AD Timer Cycle Register 3	ADCB0ADTPRR3	001F $FFFF_H$	32	FFF2 $062C_H$	3	32
ADCB0	Scan Group 3 Upper-limit / Lower-limit Table Selection Register	ADCB0ULLMSR3	00_H	8	FFF2 0630_H	3	8
ADCB0	Scan Group 4 Start Control Register	ADCB0SGSTCR4	00_H	8	FFF2 0680_H	3	8
ADCB0	AD Timer 4 Start Control Register	ADCB0ADTSTCR4	00_H	8	FFF2 0688_H	3	8
ADCB0	AD Timer 4 End Control Register	ADCB0ADTENDCR4	00_H	8	FFF2 $068C_H$	3	8
ADCB0	Scan Group 4 Control Register	ADCB0SGCR4	00_H	8	FFF2 0690_H	3	8
ADCB0	Scan Group 4 Start Virtual Channel Pointer	ADCB0SGVCSP4	00_H	8	FFF2 0694_H	3	8
ADCB0	Scan Group 4 End Virtual Channel Pointer	ADCB0SGVCEP4	00_H	8	FFF2 0698_H	3	8
ADCB0	Scan Group 4 Multi Cycle Register	ADCB0SGMCYCR4	00_H	8	FFF2 $069C_H$	3	8
ADCB0	Scan Group 4 Status Register	ADCB0SGSR4	00_H	8	FFF2 $06A4_H$	3	8
ADCB0	AD Timer Initial Phase Register 4	ADCB0ADTIPR4	0000 0000_H	32	FFF2 $06A8_H$	3	32
ADCB0	AD Timer Cycle Register 4	ADCB0ADTPRR4	001F $FFFF_H$	32	FFF2 $06AC_H$	3	32
ADCB0	Scan Group 4 Upper-limit / Lower-limit Table Selection Register	ADCB0ULLMSR4	00_H	8	FFF2 $06B0_H$	3	8
ADCB1	Virtual Channel Register 0	ADCB1VCR00	0000_H	16	FFF2 1000_H	3	8, 16
ADCB1	Virtual Channel Register 1	ADCB1VCR01	0000_H	16	FFF2 1004_H	3	8, 16
ADCB1	Virtual Channel Register 2	ADCB1VCR02	0000_H	16	FFF2 1008_H	3	8, 16
ADCB1	Virtual Channel Register 3	ADCB1VCR03	0000_H	16	FFF2 $100C_H$	3	8, 16
ADCB1	Virtual Channel Register 4	ADCB1VCR04	0000_H	16	FFF2 1010_H	3	8, 16
ADCB1	Virtual Channel Register 5	ADCB1VCR05	0000_H	16	FFF2 1014_H	3	8, 16
ADCB1	Virtual Channel Register 6	ADCB1VCR06	0000_H	16	FFF2 1018_H	3	8, 16
ADCB1	Virtual Channel Register 7	ADCB1VCR07	0000_H	16	FFF2 $101C_H$	3	8, 16
ADCB1	Virtual Channel Register 8	ADCB1VCR08	0000_H	16	FFF2 1020_H	3	8, 16
ADCB1	Virtual Channel Register 9	ADCB1VCR09	0000_H	16	FFF2 1024_H	3	8, 16
ADCB1	Virtual Channel Register 10	ADCB1VCR10	0000_H	16	FFF2 1028_H	3	8, 16
ADCB1	Virtual Channel Register 11	ADCB1VCR11	0000_H	16	FFF2 $102C_H$	3	8, 16
ADCB1	Virtual Channel Register 12	ADCB1VCR12	0000_H	16	FFF2 1030_H	3	8, 16
ADCB1	Virtual Channel Register 13	ADCB1VCR13	0000_H	16	FFF2 1034_H	3	8, 16
ADCB1	Virtual Channel Register 14	ADCB1VCR14	0000_H	16	FFF2 1038_H	3	8, 16
ADCB1	Virtual Channel Register 15	ADCB1VCR15	0000_H	16	FFF2 $103C_H$	3	8, 16
ADCB1	Virtual Channel Register 16	ADCB1VCR16	0000_H	16	FFF2 1040_H	3	8, 16
ADCB1	Virtual Channel Register 17	ADCB1VCR17	0000_H	16	FFF2 1044_H	3	8, 16
ADCB1	Virtual Channel Register 18	ADCB1VCR18	0000_H	16	FFF2 1048_H	3	8, 16
ADCB1	Virtual Channel Register 19	ADCB1VCR19	0000_H	16	FFF2 $104C_H$	3	8, 16
ADCB1	Virtual Channel Register 20	ADCB1VCR20	0000_H	16	FFF2 1050_H	3	8, 16
ADCB1	Virtual Channel Register 21	ADCB1VCR21	0000_H	16	FFF2 1054_H	3	8, 16
ADCB1	Virtual Channel Register 22	ADCB1VCR22	0000_H	16	FFF2 1058_H	3	8, 16
ADCB1	Virtual Channel Register 23	ADCB1VCR23	0000_H	16	FFF2 $105C_H$	3	8, 16
ADCB1	Virtual Channel Register 24	ADCB1VCR24	0000_H	16	FFF2 1060_H	3	8, 16
ADCB1	Virtual Channel Register 25	ADCB1VCR25	0000_H	16	FFF2 1064_H	3	8, 16
ADCB1	Virtual Channel Register 26	ADCB1VCR26	0000_H	16	FFF2 1068_H	3	8, 16
ADCB1	Virtual Channel Register 27	ADCB1VCR27	0000_H	16	FFF2 $106C_H$	3	8, 16
ADCB1	Virtual Channel Register 28	ADCB1VCR28	0000_H	16	FFF2 1070_H	3	8, 16
ADCB1	Virtual Channel Register 29	ADCB1VCR29	0000_H	16	FFF2 1074_H	3	8, 16

(120/201)

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
ADCB1	Virtual Channel Register 30	ADCB1VCR30	0000_H	16	FFF2 1078_H	3	8, 16
ADCB1	Virtual Channel Register 31	ADCB1VCR31	0000_H	16	FFF2 $107C_H$	3	8, 16
ADCB1	Virtual Channel Register 32	ADCB1VCR32	0000_H	16	FFF2 1080_H	3	8, 16
ADCB1	Virtual Channel Register 33	ADCB1VCR33	0000_H	16	FFF2 1084_H	3	8, 16
ADCB1	Virtual Channel Register 34	ADCB1VCR34	0000_H	16	FFF2 1088_H	3	8, 16
ADCB1	Virtual Channel Register 35	ADCB1VCR35	0000_H	16	FFF2 $108C_H$	3	8, 16
ADCB1	Virtual Channel Register 36	ADCB1VCR36	0000_H	16	FFF2 1090_H	3	8, 16
ADCB1	Virtual Channel Register 37	ADCB1VCR37	0000_H	16	FFF2 1094_H	3	8, 16
ADCB1	Virtual Channel Register 38	ADCB1VCR38	0000_H	16	FFF2 1098_H	3	8, 16
ADCB1	Virtual Channel Register 39	ADCB1VCR39	0000_H	16	FFF2 $109C_H$	3	8, 16
ADCB1	Data Register 0	ADCB1DR00	0000_H	16	FFF2 1100_H	3	16, 32
ADCB1	Data Register 1	ADCB1DR01	0000_H	16	FFF2 1102_H	3	16
ADCB1	Data Register 2	ADCB1DR02	0000_H	16	FFF2 1104_H	3	16, 32
ADCB1	Data Register 3	ADCB1DR03	0000_H	16	FFF2 1106_H	3	16
ADCB1	Data Register 4	ADCB1DR04	0000_H	16	FFF2 1108_H	3	16, 32
ADCB1	Data Register 5	ADCB1DR05	0000_H	16	FFF2 $110A_H$	3	16
ADCB1	Data Register 6	ADCB1DR06	0000_H	16	FFF2 $110C_H$	3	16, 32
ADCB1	Data Register 7	ADCB1DR07	0000_H	16	FFF2 $110E_H$	3	16
ADCB1	Data Register 8	ADCB1DR08	0000_H	16	FFF2 1110_H	3	16, 32
ADCB1	Data Register 9	ADCB1DR09	0000_H	16	FFF2 1112_H	3	16
ADCB1	Data Register 10	ADCB1DR10	0000_H	16	FFF2 1114_H	3	16, 32
ADCB1	Data Register 11	ADCB1DR11	0000_H	16	FFF2 1116_H	3	16
ADCB1	Data Register 12	ADCB1DR12	0000_H	16	FFF2 1118_H	3	16, 32
ADCB1	Data Register 13	ADCB1DR13	0000_H	16	FFF2 $111A_H$	3	16
ADCB1	Data Register 14	ADCB1DR14	0000_H	16	FFF2 $111C_H$	3	16, 32
ADCB1	Data Register 15	ADCB1DR15	0000_H	16	FFF2 $111E_H$	3	16
ADCB1	Data Register 16	ADCB1DR16	0000_H	16	FFF2 1120_H	3	16, 32
ADCB1	Data Register 17	ADCB1DR17	0000_H	16	FFF2 1122_H	3	16
ADCB1	Data Register 18	ADCB1DR18	0000_H	16	FFF2 1124_H	3	16, 32
ADCB1	Data Register 19	ADCB1DR19	0000_H	16	FFF2 1126_H	3	16
ADCB1	Data Register 20	ADCB1DR20	0000_H	16	FFF2 1128_H	3	16, 32
ADCB1	Data Register 21	ADCB1DR21	0000_H	16	FFF2 $112A_H$	3	16
ADCB1	Data Register 22	ADCB1DR22	0000_H	16	FFF2 $112C_H$	3	16, 32
ADCB1	Data Register 23	ADCB1DR23	0000_H	16	FFF2 $112E_H$	3	16
ADCB1	Data Register 24	ADCB1DR24	0000_H	16	FFF2 1130_H	3	16, 32
ADCB1	Data Register 25	ADCB1DR25	0000_H	16	FFF2 1132_H	3	16
ADCB1	Data Register 26	ADCB1DR26	0000_H	16	FFF2 1134_H	3	16, 32
ADCB1	Data Register 27	ADCB1DR27	0000_H	16	FFF2 1136_H	3	16
ADCB1	Data Register 28	ADCB1DR28	0000_H	16	FFF2 1138_H	3	16, 32
ADCB1	Data Register 29	ADCB1DR29	0000_H	16	FFF2 $113A_H$	3	16
ADCB1	Data Register 30	ADCB1DR30	0000_H	16	FFF2 $113C_H$	3	16, 32
ADCB1	Data Register 31	ADCB1DR31	0000_H	16	FFF2 $113E_H$	3	16
ADCB1	Data Register 32	ADCB1DR32	0000_H	16	FFF2 1140_H	3	16, 32
ADCB1	Data Register 33	ADCB1DR33	0000_H	16	FFF2 1142_H	3	16
ADCB1	Data Register 34	ADCB1DR34	0000_H	16	FFF2 1144_H	3	16, 32
ADCB1	Data Register 35	ADCB1DR35	0000_H	16	FFF2 1146_H	3	16
ADCB1	Data Register 36	ADCB1DR36	0000_H	16	FFF2 1148_H	3	16, 32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
ADCB1	Data Register 37	ADCB1DR37	0000_H	16	FFF2 $114A_H$	3	16
ADCB1	Data Register 38	ADCB1DR38	0000_H	16	FFF2 $114C_H$	3	16, 32
ADCB1	Data Register 39	ADCB1DR39	0000_H	16	FFF2 $114E_H$	3	16
ADCB1	Data Supplementary Information Register 0	ADCB1DIR00	$0000\ 0000_H$	32	FFF2 1200_H	3	32
ADCB1	Data Supplementary Information Register 1	ADCB1DIR01	$0000\ 0000_H$	32	FFF2 1204_H	3	32
ADCB1	Data Supplementary Information Register 2	ADCB1DIR02	$0000\ 0000_H$	32	FFF2 1208_H	3	32
ADCB1	Data Supplementary Information Register 3	ADCB1DIR03	$0000\ 0000_H$	32	FFF2 $120C_H$	3	32
ADCB1	Data Supplementary Information Register 4	ADCB1DIR04	$0000\ 0000_H$	32	FFF2 1210_H	3	32
ADCB1	Data Supplementary Information Register 5	ADCB1DIR05	$0000\ 0000_H$	32	FFF2 1214_H	3	32
ADCB1	Data Supplementary Information Register 6	ADCB1DIR06	$0000\ 0000_H$	32	FFF2 1218_H	3	32
ADCB1	Data Supplementary Information Register 7	ADCB1DIR07	$0000\ 0000_H$	32	FFF2 $121C_H$	3	32
ADCB1	Data Supplementary Information Register 8	ADCB1DIR08	$0000\ 0000_H$	32	FFF2 1220_H	3	32
ADCB1	Data Supplementary Information Register 9	ADCB1DIR09	$0000\ 0000_H$	32	FFF2 1224_H	3	32
ADCB1	Data Supplementary Information Register 10	ADCB1DIR10	$0000\ 0000_H$	32	FFF2 1228_H	3	32
ADCB1	Data Supplementary Information Register 11	ADCB1DIR11	$0000\ 0000_H$	32	FFF2 $122C_H$	3	32
ADCB1	Data Supplementary Information Register 12	ADCB1DIR12	$0000\ 0000_H$	32	FFF2 1230_H	3	32
ADCB1	Data Supplementary Information Register 13	ADCB1DIR13	$0000\ 0000_H$	32	FFF2 1234_H	3	32
ADCB1	Data Supplementary Information Register 14	ADCB1DIR14	$0000\ 0000_H$	32	FFF2 1238_H	3	32
ADCB1	Data Supplementary Information Register 15	ADCB1DIR15	$0000\ 0000_H$	32	FFF2 $123C_H$	3	32
ADCB1	Data Supplementary Information Register 16	ADCB1DIR16	$0000\ 0000_H$	32	FFF2 1240_H	3	32
ADCB1	Data Supplementary Information Register 17	ADCB1DIR17	$0000\ 0000_H$	32	FFF2 1244_H	3	32
ADCB1	Data Supplementary Information Register 18	ADCB1DIR18	$0000\ 0000_H$	32	FFF2 1248_H	3	32
ADCB1	Data Supplementary Information Register 19	ADCB1DIR19	$0000\ 0000_H$	32	FFF2 $124C_H$	3	32
ADCB1	Data Supplementary Information Register 20	ADCB1DIR20	$0000\ 0000_H$	32	FFF2 1250_H	3	32
ADCB1	Data Supplementary Information Register 21	ADCB1DIR21	$0000\ 0000_H$	32	FFF2 1254_H	3	32
ADCB1	Data Supplementary Information Register 22	ADCB1DIR22	$0000\ 0000_H$	32	FFF2 1258_H	3	32
ADCB1	Data Supplementary Information Register 23	ADCB1DIR23	$0000\ 0000_H$	32	FFF2 $125C_H$	3	32
ADCB1	Data Supplementary Information Register 24	ADCB1DIR24	$0000\ 0000_H$	32	FFF2 1260_H	3	32
ADCB1	Data Supplementary Information Register 25	ADCB1DIR25	$0000\ 0000_H$	32	FFF2 1264_H	3	32
ADCB1	Data Supplementary Information Register 26	ADCB1DIR26	$0000\ 0000_H$	32	FFF2 1268_H	3	32
ADCB1	Data Supplementary Information Register 27	ADCB1DIR27	$0000\ 0000_H$	32	FFF2 $126C_H$	3	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
ADCB1	Data Supplementary Information Register 28	ADCB1DIR28	0000 0000$_H$	32	FFF2 1270$_H$	3	32
ADCB1	Data Supplementary Information Register 29	ADCB1DIR29	0000 0000$_H$	32	FFF2 1274$_H$	3	32
ADCB1	Data Supplementary Information Register 30	ADCB1DIR30	0000 0000$_H$	32	FFF2 1278$_H$	3	32
ADCB1	Data Supplementary Information Register 31	ADCB1DIR31	0000 0000$_H$	32	FFF2 127C$_H$	3	32
ADCB1	Data Supplementary Information Register 32	ADCB1DIR32	0000 0000$_H$	32	FFF2 1280$_H$	3	32
ADCB1	Data Supplementary Information Register 33	ADCB1DIR33	0000 0000$_H$	32	FFF2 1284$_H$	3	32
ADCB1	Data Supplementary Information Register 34	ADCB1DIR34	0000 0000$_H$	32	FFF2 1288$_H$	3	32
ADCB1	Data Supplementary Information Register 35	ADCB1DIR35	0000 0000$_H$	32	FFF2 128C$_H$	3	32
ADCB1	Data Supplementary Information Register 36	ADCB1DIR36	0000 0000$_H$	32	FFF2 1290$_H$	3	32
ADCB1	Data Supplementary Information Register 37	ADCB1DIR37	0000 0000$_H$	32	FFF2 1294$_H$	3	32
ADCB1	Data Supplementary Information Register 38	ADCB1DIR38	0000 0000$_H$	32	FFF2 1298$_H$	3	32
ADCB1	Data Supplementary Information Register 39	ADCB1DIR39	0000 0000$_H$	32	FFF2 129C$_H$	3	32
ADCB1	AD End Register	ADCB1ADHALTR	00$_H$	8	FFF2 1380$_H$	3	8
ADCB1	AD Control Register 1	ADCB1ADCR1	00$_H$	8	FFF2 1384$_H$	3	8
ADCB1	MPX Current Control Register	ADCB1MPXCURCR	00$_H$	8	FFF2 1388$_H$	3	8
ADCB1	MPX Current Register	ADCB1MPXCURR	0000 0000$_H$	32	FFF2 138C$_H$	3	32
ADCB1	MPX Optional Register	ADCB1MPXOWR	00$_H$	8	FFF2 1390$_H$	3	8
ADCB1	MPX Command Information Register	ADCB1MPXCMDR	00$_H$	8	FFF2 1394$_H$	3	8
ADCB1	AD Control Register 2	ADCB1ADCR2	00$_H$	8	FFF2 1398$_H$	3	8
ADCB1	DFE/ASF Entry Scan Group Enable Register	ADCB1DFASENTSGER	0000$_H$	16	FFF2 139C$_H$	3	16
ADCB1	AD Change Monitor Virtual Channel Pointer	ADCB1ADENDP	00$_H$	8	FFF2 13A0$_H$	3	8
ADCB1	Safety Control Register	ADCB1SFTCR	00$_H$	8	FFF2 13C0$_H$	3	8
ADCB1	Pin Level Self-diagnostic Control Register	ADCB1TDCR	00$_H$	8	FFF2 13C4$_H$	3	8
ADCB1	Wiring-break Detection Control Register	ADCB1ODCR	00$_H$	8	FFF2 13C8$_H$	3	8
ADCB1	Upper-Limit / Lower-Limit Table Register 0	ADCB1ULLMTBR0	7FFE 0000$_H$	32	FFF2 13CC$_H$	3	16, 32
ADCB1	Upper-Limit / Lower-Limit Table Register 1	ADCB1ULLMTBR1	7FFE 0000$_H$	32	FFF2 13D0$_H$	3	16, 32
ADCB1	Upper-Limit / Lower-Limit Table Register 2	ADCB1ULLMTBR2	7FFE 0000$_H$	32	FFF2 13D4$_H$	3	16, 32
ADCB1	Error Clear Register	ADCB1ECR	00$_H$	8	FFF2 13D8$_H$	3	8
ADCB1	Upper-Limit / Lower-Limit Error Register	ADCB1ULER	00$_H$	8	FFF2 13DC$_H$	3	8
ADCB1	Overwrite Error Register	ADCB1OWER	00$_H$	8	FFF2 13E0$_H$	3	8
ADCB1	Parity Error Register	ADCB1PER	00$_H$	8	FFF2 13E4$_H$	3	8
ADCB1	ID Error Register	ADCB1IDER	00$_H$	8	FFF2 13E8$_H$	3	8
ADCB1	Scan Group 0 Start Control Register	ADCB1SGSTCR0	00$_H$	8	FFF2 1480$_H$	3	8
ADCB1	Scan Group 0 Control Register	ADCB1SGCR0	00$_H$	8	FFF2 1490$_H$	3	8
ADCB1	Scan Group 0 Start Virtual Channel Pointer	ADCB1SGVCSP0	00$_H$	8	FFF2 1494$_H$	3	8
ADCB1	Scan Group 0 End Virtual Channel Pointer	ADCB1SGVCEP0	00$_H$	8	FFF2 1498$_H$	3	8
ADCB1	Scan Group 0 Multi Cycle Register	ADCB1SGMCYCR0	00$_H$	8	FFF2 149C$_H$	3	8

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
ADCB1	Scan Group 0 Status Register	ADCB1SGSR0	00_H	8	FFF2 $14A4_H$	3	8
ADCB1	Scan Group 0 Upper-limit / Lower-limit Table Selection Register	ADCB1ULLMSR0	00_H	8	FFF2 $14B0_H$	3	8
ADCB1	Scan Group 1 Start Control Register	ADCB1SGSTCR1	00_H	8	FFF2 1500_H	3	8
ADCB1	Scan Group 1 Control Register	ADCB1SGCR1	00_H	8	FFF2 1510_H	3	8
ADCB1	Scan Group 1 Start Virtual Channel Pointer	ADCB1SGVCSP1	00_H	8	FFF2 1514_H	3	8
ADCB1	Scan Group 1 End Virtual Channel Pointer	ADCB1SGVCEP1	00_H	8	FFF2 1518_H	3	8
ADCB1	Scan Group 1 Multi Cycle Register	ADCB1SGMCYCR1	00_H	8	FFF2 $151C_H$	3	8
ADCB1	Scan Group 1 Status Register	ADCB1SGSR1	00_H	8	FFF2 1524_H	3	8
ADCB1	Scan Group 1 Upper-limit / Lower-limit Table Selection Register	ADCB1ULLMSR1	00_H	8	FFF2 1530_H	3	8
ADCB1	Scan Group 2 Start Control Register	ADCB1SGSTCR2	00_H	8	FFF2 1580_H	3	8
ADCB1	Scan Group 2 Control Register	ADCB1SGCR2	00_H	8	FFF2 1590_H	3	8
ADCB1	Scan Group 2 Start Virtual Channel Pointer	ADCB1SGVCSP2	00_H	8	FFF2 1594_H	3	8
ADCB1	Scan Group 2 End Virtual Channel Pointer	ADCB1SGVCEP2	00_H	8	FFF2 1598_H	3	8
ADCB1	Scan Group 2 Multi Cycle Register	ADCB1SGMCYCR2	00_H	8	FFF2 $159C_H$	3	8
ADCB1	Scan Group 2 Status Register	ADCB1SGSR2	00_H	8	FFF2 $15A4_H$	3	8
ADCB1	Scan Group 2 Upper-limit / Lower-limit Table Selection Register	ADCB1ULLMSR2	00_H	8	FFF2 $15B0_H$	3	8
ADCB1	Scan Group 3 Start Control Register	ADCB1SGSTCR3	00_H	8	FFF2 1600_H	3	8
ADCB1	AD Timer 3 Start Control Register	ADCB1ADTSTCR3	00_H	8	FFF2 1608_H	3	8
ADCB1	AD Timer 3 End Control Register	ADCB1ADTENDCR3	00_H	8	FFF2 $160C_H$	3	8
ADCB1	Scan Group 3 Control Register	ADCB1SGCR3	00_H	8	FFF2 1610_H	3	8
ADCB1	Scan Group 3 Start Virtual Channel Pointer	ADCB1SGVCSP3	00_H	8	FFF2 1614_H	3	8
ADCB1	Scan Group 3 End Virtual Channel Pointer	ADCB1SGVCEP3	00_H	8	FFF2 1618_H	3	8
ADCB1	Scan Group 3 Multi Cycle Register	ADCB1SGMCYCR3	00_H	8	FFF2 $161C_H$	3	8
ADCB1	Scan Group 3 Status Register	ADCB1SGSR3	00_H	8	FFF2 1624_H	3	8
ADCB1	AD Timer Initial Phase Register 3	ADCB1ADTIPR3	0000 0000_H	32	FFF2 1628_H	3	32
ADCB1	AD Timer Cycle Register 3	ADCB1ADTPRR3	001F $FFFF_H$	32	FFF2 $162C_H$	3	32
ADCB1	Scan Group 3 Upper-limit / Lower-limit Table Selection Register	ADCB1ULLMSR3	00_H	8	FFF2 1630_H	3	8
ADCB1	Scan Group 4 Start Control Register	ADCB1SGSTCR4	00_H	8	FFF2 1680_H	3	8
ADCB1	AD Timer 4 Start Control Register	ADCB1ADTSTCR4	00_H	8	FFF2 1688_H	3	8
ADCB1	AD Timer 4 End Control Register	ADCB1ADTENDCR4	00_H	8	FFF2 $168C_H$	3	8
ADCB1	Scan Group 4 Control Register	ADCB1SGCR4	00_H	8	FFF2 1690_H	3	8
ADCB1	Scan Group 4 Start Virtual Channel Pointer	ADCB1SGVCSP4	00_H	8	FFF2 1694_H	3	8
ADCB1	Scan Group 4 End Virtual Channel Pointer	ADCB1SGVCEP4	00_H	8	FFF2 1698_H	3	8
ADCB1	Scan Group 4 Multi Cycle Register	ADCB1SGMCYCR4	00_H	8	FFF2 $169C_H$	3	8
ADCB1	Scan Group 4 Status Register	ADCB1SGSR4	00_H	8	FFF2 $16A4_H$	3	8
ADCB1	AD Timer Initial Phase Register 4	ADCB1ADTIPR4	0000 0000_H	32	FFF2 $16A8_H$	3	32
ADCB1	AD Timer Cycle Register 4	ADCB1ADTPRR4	001F $FFFF_H$	32	FFF2 $16AC_H$	3	32
ADCB1	Scan Group 4 Upper-limit / Lower-limit Table Selection Register	ADCB1ULLMSR4	00_H	8	FFF2 $16B0_H$	3	8
ASF10	Accumulation Data Read Register 0	ASF10DR00	0000 0000_H	32	FFF2 2000_H	3	32

(124/201)

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
ASF10	Accumulation Data Read Register 1	ASF10DR01	0000 0000$_H$	32	FFF2 2004$_H$	3	32
ASF10	Accumulation Data Read Register 2	ASF10DR02	0000 0000$_H$	32	FFF2 2008$_H$	3	32
ASF10	Accumulation Data Read Register 3	ASF10DR03	0000 0000$_H$	32	FFF2 200C$_H$	3	32
ASF10	Accumulation Data Read Register 4	ASF10DR04	0000 0000$_H$	32	FFF2 2010$_H$	3	32
ASF10	Accumulation Data Read Register 5	ASF10DR05	0000 0000$_H$	32	FFF2 2014$_H$	3	32
ASF10	Accumulation Data Read Register 6	ASF10DR06	0000 0000$_H$	32	FFF2 2018$_H$	3	32
ASF10	Accumulation Data Read Register 7	ASF10DR07	0000 0000$_H$	32	FFF2 201C$_H$	3	32
ASF10	Accumulation Data Read Register 8	ASF10DR08	0000 0000$_H$	32	FFF2 2020$_H$	3	32
ASF10	Accumulation Data Read Register 9	ASF10DR09	0000 0000$_H$	32	FFF2 2024$_H$	3	32
ASF10	Accumulation Data Read Register 10	ASF10DR10	0000 0000$_H$	32	FFF2 2028$_H$	3	32
ASF10	Accumulation Compare Match Register 0	ASF10CMP00	00$_H$	8	FFF2 2040$_H$	3	8
ASF10	Accumulation Compare Match Register 1	ASF10CMP01	00$_H$	8	FFF2 2044$_H$	3	8
ASF10	Accumulation Compare Match Register 2	ASF10CMP02	00$_H$	8	FFF2 2048$_H$	3	8
ASF10	Accumulation Compare Match Register 3	ASF10CMP03	00$_H$	8	FFF2 204C$_H$	3	8
ASF10	Accumulation Compare Match Register 4	ASF10CMP04	00$_H$	8	FFF2 2050$_H$	3	8
ASF10	Accumulation Compare Match Register 5	ASF10CMP05	00$_H$	8	FFF2 2054$_H$	3	8
ASF10	Accumulation Compare Match Register 6	ASF10CMP06	00$_H$	8	FFF2 2058$_H$	3	8
ASF10	Accumulation Compare Match Register 7	ASF10CMP07	00$_H$	8	FFF2 205C$_H$	3	8
ASF10	Accumulation Compare Match Register 8	ASF10CMP08	00$_H$	8	FFF2 2060$_H$	3	8
ASF10	Accumulation Compare Match Register 9	ASF10CMP09	00$_H$	8	FFF2 2064$_H$	3	8
ASF10	Accumulation Compare Match Register 10	ASF10CMP10	00$_H$	8	FFF2 2068$_H$	3	8
ASF10	Accumulation Counter Control Register 0	ASF10CTL0	0000$_H$	16	FFF2 2080$_H$	3	8, 16
ASF10	Accumulation Counter Control Register 1	ASF10CTL1	00$_H$	8	FFF2 2084$_H$	3	8
ASF10	Accumulation Count Read Register	ASF10CNT	00$_H$	8	FFF2 208C$_H$	3	8
IFC	Floating-point Data Register 000	FDR000	0000 0000$_H$	32	FFF2 3000$_H$	3	32
IFC	Floating-point Data Register 001	FDR001	0000 0000$_H$	32	FFF2 3004$_H$	3	32
IFC	Floating-point Data Register 002	FDR002	0000 0000$_H$	32	FFF2 3008$_H$	3	32
IFC	Floating-point Data Register 003	FDR003	0000 0000$_H$	32	FFF2 300C$_H$	3	32
IFC	Floating-point Data Register 004	FDR004	0000 0000$_H$	32	FFF2 3010$_H$	3	32
IFC	Floating-point Data Register 005	FDR005	0000 0000$_H$	32	FFF2 3014$_H$	3	32
IFC	Floating-point Data Register 006	FDR006	0000 0000$_H$	32	FFF2 3018$_H$	3	32
IFC	Floating-point Data Register 007	FDR007	0000 0000$_H$	32	FFF2 301C$_H$	3	32
IFC	Floating-point Data Register 008	FDR008	0000 0000$_H$	32	FFF2 3020$_H$	3	32
IFC	Floating-point Data Register 009	FDR009	0000 0000$_H$	32	FFF2 3024$_H$	3	32
IFC	Floating-point Data Register 010	FDR010	0000 0000$_H$	32	FFF2 3028$_H$	3	32
IFC	Floating-point Data Register 011	FDR011	0000 0000$_H$	32	FFF2 302C$_H$	3	32
IFC	Floating-point Data Register 012	FDR012	0000 0000$_H$	32	FFF2 3030$_H$	3	32
IFC	Floating-point Data Register 013	FDR013	0000 0000$_H$	32	FFF2 3034$_H$	3	32
IFC	Floating-point Data Register 014	FDR014	0000 0000$_H$	32	FFF2 3038$_H$	3	32
IFC	Floating-point Data Register 015	FDR015	0000 0000$_H$	32	FFF2 303C$_H$	3	32
IFC	Floating-point Data Register 016	FDR016	0000 0000$_H$	32	FFF2 3040$_H$	3	32
IFC	Floating-point Data Register 017	FDR017	0000 0000$_H$	32	FFF2 3044$_H$	3	32
IFC	Floating-point Data Register 018	FDR018	0000 0000$_H$	32	FFF2 3048$_H$	3	32
IFC	Floating-point Data Register 019	FDR019	0000 0000$_H$	32	FFF2 304C$_H$	3	32
IFC	Floating-point Data Register 020	FDR020	0000 0000$_H$	32	FFF2 3050$_H$	3	32
IFC	Floating-point Data Register 021	FDR021	0000 0000$_H$	32	FFF2 3054$_H$	3	32
IFC	Floating-point Data Register 022	FDR022	0000 0000$_H$	32	FFF2 3058$_H$	3	32

(125/201)

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
IFC	Floating-point Data Register 023	FDR023	0000 0000$_H$	32	FFF2 305C$_H$	3	32
IFC	Floating-point Data Register 024	FDR024	0000 0000$_H$	32	FFF2 3060$_H$	3	32
IFC	Floating-point Data Register 025	FDR025	0000 0000$_H$	32	FFF2 3064$_H$	3	32
IFC	Floating-point Data Register 026	FDR026	0000 0000$_H$	32	FFF2 3068$_H$	3	32
IFC	Floating-point Data Register 027	FDR027	0000 0000$_H$	32	FFF2 306C$_H$	3	32
IFC	Floating-point Data Register 028	FDR028	0000 0000$_H$	32	FFF2 3070$_H$	3	32
IFC	Floating-point Data Register 029	FDR029	0000 0000$_H$	32	FFF2 3074$_H$	3	32
IFC	Floating-point Data Register 030	FDR030	0000 0000$_H$	32	FFF2 3078$_H$	3	32
IFC	Floating-point Data Register 031	FDR031	0000 0000$_H$	32	FFF2 307C$_H$	3	32
IFC	Floating-point Data Register 032	FDR032	0000 0000$_H$	32	FFF2 3080$_H$	3	32
IFC	Floating-point Data Register 033	FDR033	0000 0000$_H$	32	FFF2 3084$_H$	3	32
IFC	Floating-point Data Register 034	FDR034	0000 0000$_H$	32	FFF2 3088$_H$	3	32
IFC	Floating-point Data Register 035	FDR035	0000 0000$_H$	32	FFF2 308C$_H$	3	32
IFC	Floating-point Data Register 036	FDR036	0000 0000$_H$	32	FFF2 3090$_H$	3	32
IFC	Floating-point Data Register 037	FDR037	0000 0000$_H$	32	FFF2 3094$_H$	3	32
IFC	Floating-point Data Register 038	FDR038	0000 0000$_H$	32	FFF2 3098$_H$	3	32
IFC	Floating-point Data Register 039	FDR039	0000 0000$_H$	32	FFF2 309C$_H$	3	32
IFC	Floating-point Data Register 100	FDR100	0000 0000$_H$	32	FFF2 3100$_H$	3	32
IFC	Floating-point Data Register 101	FDR101	0000 0000$_H$	32	FFF2 3104$_H$	3	32
IFC	Floating-point Data Register 102	FDR102	0000 0000$_H$	32	FFF2 3108$_H$	3	32
IFC	Floating-point Data Register 103	FDR103	0000 0000$_H$	32	FFF2 310C$_H$	3	32
IFC	Floating-point Data Register 104	FDR104	0000 0000$_H$	32	FFF2 3110$_H$	3	32
IFC	Floating-point Data Register 105	FDR105	0000 0000$_H$	32	FFF2 3114$_H$	3	32
IFC	Floating-point Data Register 106	FDR106	0000 0000$_H$	32	FFF2 3118$_H$	3	32
IFC	Floating-point Data Register 107	FDR107	0000 0000$_H$	32	FFF2 311C$_H$	3	32
IFC	Floating-point Data Register 108	FDR108	0000 0000$_H$	32	FFF2 3120$_H$	3	32
IFC	Floating-point Data Register 109	FDR109	0000 0000$_H$	32	FFF2 3124$_H$	3	32
IFC	Floating-point Data Register 110	FDR110	0000 0000$_H$	32	FFF2 3128$_H$	3	32
IFC	Floating-point Data Register 111	FDR111	0000 0000$_H$	32	FFF2 312C$_H$	3	32
IFC	Floating-point Data Register 112	FDR112	0000 0000$_H$	32	FFF2 3130$_H$	3	32
IFC	Floating-point Data Register 113	FDR113	0000 0000$_H$	32	FFF2 3134$_H$	3	32
IFC	Floating-point Data Register 114	FDR114	0000 0000$_H$	32	FFF2 3138$_H$	3	32
IFC	Floating-point Data Register 115	FDR115	0000 0000$_H$	32	FFF2 313C$_H$	3	32
IFC	Floating-point Data Register 116	FDR116	0000 0000$_H$	32	FFF2 3140$_H$	3	32
IFC	Floating-point Data Register 117	FDR117	0000 0000$_H$	32	FFF2 3144$_H$	3	32
IFC	Floating-point Data Register 118	FDR118	0000 0000$_H$	32	FFF2 3148$_H$	3	32
IFC	Floating-point Data Register 119	FDR119	0000 0000$_H$	32	FFF2 314C$_H$	3	32
IFC	Floating-point Data Register 120	FDR120	0000 0000$_H$	32	FFF2 3150$_H$	3	32
IFC	Floating-point Data Register 121	FDR121	0000 0000$_H$	32	FFF2 3154$_H$	3	32
IFC	Floating-point Data Register 122	FDR122	0000 0000$_H$	32	FFF2 3158$_H$	3	32
IFC	Floating-point Data Register 123	FDR123	0000 0000$_H$	32	FFF2 315C$_H$	3	32
IFC	Floating-point Data Register 124	FDR124	0000 0000$_H$	32	FFF2 3160$_H$	3	32
IFC	Floating-point Data Register 125	FDR125	0000 0000$_H$	32	FFF2 3164$_H$	3	32
IFC	Floating-point Data Register 126	FDR126	0000 0000$_H$	32	FFF2 3168$_H$	3	32
IFC	Floating-point Data Register 127	FDR127	0000 0000$_H$	32	FFF2 316C$_H$	3	32
IFC	Floating-point Data Register 128	FDR128	0000 0000$_H$	32	FFF2 3170$_H$	3	32
IFC	Floating-point Data Register 129	FDR129	0000 0000$_H$	32	FFF2 3174$_H$	3	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
IFC	Floating-point Data Register 130	FDR130	0000 0000$_H$	32	FFF2 3178$_H$	3	32
IFC	Floating-point Data Register 131	FDR131	0000 0000$_H$	32	FFF2 317C$_H$	3	32
IFC	Floating-point Data Register 132	FDR132	0000 0000$_H$	32	FFF2 3180$_H$	3	32
IFC	Floating-point Data Register 133	FDR133	0000 0000$_H$	32	FFF2 3184$_H$	3	32
IFC	Floating-point Data Register 134	FDR134	0000 0000$_H$	32	FFF2 3188$_H$	3	32
IFC	Floating-point Data Register 135	FDR135	0000 0000$_H$	32	FFF2 318C$_H$	3	32
IFC	Floating-point Data Register 136	FDR136	0000 0000$_H$	32	FFF2 3190$_H$	3	32
IFC	Floating-point Data Register 137	FDR137	0000 0000$_H$	32	FFF2 3194$_H$	3	32
IFC	Floating-point Data Register 138	FDR138	0000 0000$_H$	32	FFF2 3198$_H$	3	32
IFC	Floating-point Data Register 139	FDR139	0000 0000$_H$	32	FFF2 319C$_H$	3	32
DSADCC	AD Synchronization Start Control Register	DSADCCADSYNSTCR	00$_H$	8	FFF3 0000$_H$	3	8
DSADCC	Over Sampling Rate Control Register	DSADCCOSMPRCR	00$_H$	8	FFF3 0004$_H$	3	8
DSADCC	Pin Level Self-diagnostic Control Register	DSADCCTDCR	00$_H$	8	FFF3 0008$_H$	3	8
DSADCC	Pin Level Self-diagnostic Level Setting Register	DSADCCTDLVR	0000$_H$	16	FFF3 000C$_H$	3	8, 16
DSADC0	Channel Control Register	DSADC0CCR	0000 0000$_H$	32	FFF3 1000$_H$	3	8, 16, 32
DSADC0	Data Supplementary Information Register	DSADC0DIR	0000 0000$_H$	32	FFF3 1004$_H$	3	32
DSADC0	AD Start Control Register	DSADC0ADSTCR	00$_H$	8	FFF3 1008$_H$	3	8
DSADC0	AD Stop Control Register	DSADC0ADENDCR	00$_H$	8	FFF3 100C$_H$	3	8
DSADC0	AD Control Register	DSADC0ADCR	00$_H$	8	FFF3 1010$_H$	3	8
DSADC0	AD Status Register	DSADC0ADSR	00$_H$	8	FFF3 1014$_H$	3	8
DSADC0	Safety Control Register	DSADC0SFTCR	00$_H$	8	FFF3 1018$_H$	3	8
DSADC0	Upper-Limit / Lower-Limit Table Register	DSADC0ULLMTBR	7FFF 8000$_H$	32	FFF3 101C$_H$	3	16, 32
DSADC0	Error Clear Register	DSADC0ECR	00$_H$	8	FFF3 1020$_H$	3	8
DSADC0	Error Register	DSADC0ER	00$_H$	8	FFF3 1024$_H$	3	8
DSADC0	Digital Filter FIR Control Register	DSADC0FCR	00$_H$	8	FFF3 1028$_H$	3	8
DSADC1	Channel Control Register	DSADC1CCR	0000 0000$_H$	32	FFF3 2000$_H$	3	8, 16, 32
DSADC1	Data Supplementary Information Register	DSADC1DIR	0000 0000$_H$	32	FFF3 2004$_H$	3	32
DSADC1	AD Start Control Register	DSADC1ADSTCR	00$_H$	8	FFF3 2008$_H$	3	8
DSADC1	AD Stop Control Register	DSADC1ADENDCR	00$_H$	8	FFF3 200C$_H$	3	8
DSADC1	AD Control Register	DSADC1ADCR	00$_H$	8	FFF3 2010$_H$	3	8
DSADC1	AD Status Register	DSADC1ADSR	00$_H$	8	FFF3 2014$_H$	3	8
DSADC1	Safety Control Register	DSADC1SFTCR	00$_H$	8	FFF3 2018$_H$	3	8
DSADC1	Upper-Limit / Lower-Limit Table Register	DSADC1ULLMTBR	7FFF 8000$_H$	32	FFF3 201C$_H$	3	16, 32
DSADC1	Error Clear Register	DSADC1ECR	00$_H$	8	FFF3 2020$_H$	3	8
DSADC1	Error Register	DSADC1ER	00$_H$	8	FFF3 2024$_H$	3	8
DSADC1	Digital Filter FIR Control Register	DSADC1FCR	00$_H$	8	FFF3 2028$_H$	3	8
DSADC2	Channel Control Register	DSADC2CCR	0000 0000$_H$	32	FFF3 3000$_H$	3	8, 16, 32
DSADC2	Data Supplementary Information Register	DSADC2DIR	0000 0000$_H$	32	FFF3 3004$_H$	3	32
DSADC2	AD Start Control Register	DSADC2ADSTCR	00$_H$	8	FFF3 3008$_H$	3	8
DSADC2	AD Stop Control Register	DSADC2ADENDCR	00$_H$	8	FFF3 300C$_H$	3	8
DSADC2	AD Control Register	DSADC2ADCR	00$_H$	8	FFF3 3010$_H$	3	8
DSADC2	AD Status Register	DSADC2ADSR	00$_H$	8	FFF3 3014$_H$	3	8
DSADC2	Safety Control Register	DSADC2SFTCR	00$_H$	8	FFF3 3018$_H$	3	8

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
DSADC2	Upper-Limit / Lower-Limit Table Register	DSADC2ULLMTBR	7FFF 8000$_H$	32	FFF3 301C$_H$	3	16, 32
DSADC2	Error Clear Register	DSADC2ECR	00$_H$	8	FFF3 3020$_H$	3	8
DSADC2	Error Register	DSADC2ER	00$_H$	8	FFF3 3024$_H$	3	8
DSADC2	Digital Filter FIR Control Register	DSADC2FCR	00$_H$	8	FFF3 3028$_H$	3	8
DSADC3	Channel Control Register	DSADC3CCR	0000 0000$_H$	32	FFF3 4000$_H$	3	8, 16, 32
DSADC3	Data Supplementary Information Register	DSADC3DIR	0000 0000$_H$	32	FFF3 4004$_H$	3	32
DSADC3	AD Start Control Register	DSADC3ADSTCR	00$_H$	8	FFF3 4008$_H$	3	8
DSADC3	AD Stop Control Register	DSADC3ADENDCR	00$_H$	8	FFF3 400C$_H$	3	8
DSADC3	AD Control Register	DSADC3ADCR	00$_H$	8	FFF3 4010$_H$	3	8
DSADC3	AD Status Register	DSADC3ADSR	00$_H$	8	FFF3 4014$_H$	3	8
DSADC3	Safety Control Register	DSADC3SFTCR	00$_H$	8	FFF3 4018$_H$	3	8
DSADC3	Upper-Limit / Lower-Limit Table Register	DSADC3ULLMTBR	7FFF 8000$_H$	32	FFF3 401C$_H$	3	16, 32
DSADC3	Error Clear Register	DSADC3ECR	00$_H$	8	FFF3 4020$_H$	3	8
DSADC3	Error Register	DSADC3ER	00$_H$	8	FFF3 4024$_H$	3	8
DSADC3	Digital Filter FIR Control Register	DSADC3FCR	00$_H$	8	FFF3 4028$_H$	3	8
DSADC4	Channel Control Register	DSADC4CCR	0000 0000$_H$	32	FFF3 5000$_H$	3	8, 16, 32
DSADC4	Data Supplementary Information Register	DSADC4DIR	0000 0000$_H$	32	FFF3 5004$_H$	3	32
DSADC4	AD Start Control Register	DSADC4ADSTCR	00$_H$	8	FFF3 5008$_H$	3	8
DSADC4	AD Stop Control Register	DSADC4ADENDCR	00$_H$	8	FFF3 500C$_H$	3	8
DSADC4	AD Control Register	DSADC4ADCR	00$_H$	8	FFF3 5010$_H$	3	8
DSADC4	AD Status Register	DSADC4ADSR	00$_H$	8	FFF3 5014$_H$	3	8
DSADC4	Safety Control Register	DSADC4SFTCR	00$_H$	8	FFF3 5018$_H$	3	8
DSADC4	Upper-Limit / Lower-Limit Table Register	DSADC4ULLMTBR	7FFF 8000$_H$	32	FFF3 501C$_H$	3	16, 32
DSADC4	Error Clear Register	DSADC4ECR	00$_H$	8	FFF3 5020$_H$	3	8
DSADC4	Error Register	DSADC4ER	00$_H$	8	FFF3 5024$_H$	3	8
DSADC4	Digital Filter FIR Control Register	DSADC4FCR	00$_H$	8	FFF3 5028$_H$	3	8
DSADC5	Channel Control Register	DSADC5CCR	0000 0000$_H$	32	FFF3 6000$_H$	3	8, 16, 32
DSADC5	Data Supplementary Information Register	DSADC5DIR	0000 0000$_H$	32	FFF3 6004$_H$	3	32
DSADC5	AD Start Control Register	DSADC5ADSTCR	00$_H$	8	FFF3 6008$_H$	3	8
DSADC5	AD Stop Control Register	DSADC5ADENDCR	00$_H$	8	FFF3 600C$_H$	3	8
DSADC5	AD Control Register	DSADC5ADCR	00$_H$	8	FFF3 6010$_H$	3	8
DSADC5	AD Status Register	DSADC5ADSR	00$_H$	8	FFF3 6014$_H$	3	8
DSADC5	Safety Control Register	DSADC5SFTCR	00$_H$	8	FFF3 6018$_H$	3	8
DSADC5	Upper-Limit / Lower-Limit Table Register	DSADC5ULLMTBR	7FFF 8000$_H$	32	FFF3 601C$_H$	3	16, 32
DSADC5	Error Clear Register	DSADC5ECR	00$_H$	8	FFF3 6020$_H$	3	8
DSADC5	Error Register	DSADC5ER	00$_H$	8	FFF3 6024$_H$	3	8
DSADC5	Digital Filter FIR Control Register	DSADC5FCR	00$_H$	8	FFF3 6028$_H$	3	8
DSADC6	Channel Control Register	DSADC6CCR	0000 0000$_H$	32	FFF3 7000$_H$	3	8, 16, 32
DSADC6	Data Supplementary Information Register	DSADC6DIR	0000 0000$_H$	32	FFF3 7004$_H$	3	32
DSADC6	AD Start Control Register	DSADC6ADSTCR	00$_H$	8	FFF3 7008$_H$	3	8
DSADC6	AD Stop Control Register	DSADC6ADENDCR	00$_H$	8	FFF3 700C$_H$	3	8
DSADC6	AD Control Register	DSADC6ADCR	00$_H$	8	FFF3 7010$_H$	3	8
DSADC6	AD Status Register	DSADC6ADSR	00$_H$	8	FFF3 7014$_H$	3	8
DSADC6	Safety Control Register	DSADC6SFTCR	00$_H$	8	FFF3 7018$_H$	3	8

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
DSADC6	Upper-Limit / Lower-Limit Table Register	DSADC6ULLMTBR	7FFF 8000$_H$	32	FFF3 701C$_H$	3	16, 32
DSADC6	Error Clear Register	DSADC6ECR	00$_H$	8	FFF3 7020$_H$	3	8
DSADC6	Error Register	DSADC6ER	00$_H$	8	FFF3 7024$_H$	3	8
DSADC6	Digital Filter FIR Control Register	DSADC6FCR	00$_H$	8	FFF3 7028$_H$	3	8
DSADC7	Channel Control Register	DSADC7CCR	0000 0000$_H$	32	FFF3 8000$_H$	3	8, 16, 32
DSADC7	Data Supplementary Information Register	DSADC7DIR	0000 0000$_H$	32	FFF3 8004$_H$	3	32
DSADC7	AD Start Control Register	DSADC7ADSTCR	00$_H$	8	FFF3 8008$_H$	3	8
DSADC7	AD Stop Control Register	DSADC7ADENDCR	00$_H$	8	FFF3 800C$_H$	3	8
DSADC7	AD Control Register	DSADC7ADCR	00$_H$	8	FFF3 8010$_H$	3	8
DSADC7	AD Status Register	DSADC7ADSR	00$_H$	8	FFF3 8014$_H$	3	8
DSADC7	Safety Control Register	DSADC7SFTCR	00$_H$	8	FFF3 8018$_H$	3	8
DSADC7	Upper-Limit / Lower-Limit Table Register	DSADC7ULLMTBR	7FFF 8000$_H$	32	FFF3 801C$_H$	3	16, 32
DSADC7	Error Clear Register	DSADC7ECR	00$_H$	8	FFF3 8020$_H$	3	8
DSADC7	Error Register	DSADC7ER	00$_H$	8	FFF3 8024$_H$	3	8
DSADC7	Digital Filter FIR Control Register	DSADC7FCR	00$_H$	8	FFF3 8028$_H$	3	8
DCRA0	CRC Input Register	DCRA0CIN	0000 0000$_H$	32	FFF7 0000$_H$	3	32
DCRA0	CRC Data Register	DCRA0COUT	0000 0000$_H$	32	FFF7 0004$_H$	3	32
DCRA0	CRC Control Register	DCRA0CTL	00$_H$	8	FFF7 0020$_H$	3	8
FLASH	FHVE3 Control Register	FHVE3	0000 0000$_H$	32	FFF8 2410$_H$	5	32
SYS	Reset Source Determination Register	RESF	0000 0000$_H$	32	FFF8 2800$_H$	5	32
SYS	Reset Source Clear Register	RESFC	0000 0000$_H$	32	FFF8 2808$_H$	5	32
SYS	CVM Detection Flag Register	DETFLG	0000 0000$_H$	32	FFF8 2820$_H$	5	32
SYS	CVM Detection Flag Clear Register	DETFLGC	0000 0000$_H$	32	FFF8 2828$_H$	5	32
SYS	EPT Control Register	EPTCNT	0000 0000$_H$	32	FFF8 2C0C$_H$	5	32
SYS	CVM Control Register	VSCTL	0000 0000$_H$	32	FFF8 2C10$_H$	5	32
SYS	Upper Limit Voltage Setting Register	HDETCTL	0000 0003$_H$	32	FFF8 2C14$_H$	5	32
SYS	Lower Limit Voltage Setting Register	LDETCTL	0000 0003$_H$	32	FFF8 2C18$_H$	5	32
SYS	Detection Signal Filter Control Register	VSDETFCTL	0000 0000$_H$	32	FFF8 2C1C$_H$	5	32
SYS	Protection Command Register	PROT0PHCMD	0000 0000$_H$	32	FFF8 3000$_H$	5	32
SYS	Protection Command Status Register	PROT0PS	0000 0000$_H$	32	FFF8 3004$_H$	5	32
SYS	PLL0 Status Register	PLL0CLKS	0000 0001$_H$	32	FFF8 8004$_H$	5	32
SYS	PLL0 Control Register 1	PLL0CLKC1	0000 0000$_H$	32	FFF8 8200$_H$	5	32
CLMAC	CLMA Self Test Register	CLMATEST	0000 0000$_H$	32	FFF8 8204$_H$	5	32
CLMAC	CLMA Self Test Status Register	CLMATESTS	0000 0000$_H$	32	FFF8 8208$_H$	5	32
CLMA0	CLMA0 Control Register 0	CLMA0CTL0	00$_H$	8	FFF8 8400$_H$	5	8
CLMA0	CLMA0 compare Register L	CLMA0CMPL	0001$_H$	16	FFF8 8408$_H$	5	16
CLMA0	CLMA0 compare Register H	CLMA0CMPH	03FF$_H$	16	FFF8 840C$_H$	5	16
CLMA0	CLMA0 Protection instruction Register	CLMA0PCMD	00$_H$	8	FFF8 8410$_H$	5	8
CLMA0	CLMA0 Protection Status Register	CLMA0PS	00$_H$	8	FFF8 8414$_H$	5	8
CLMA1	CLMA1 Control Register 0	CLMA1CTL0	00$_H$	8	FFF8 8420$_H$	5	8
CLMA1	CLMA1 Compare Register L	CLMA1CMPL	0001$_H$	16	FFF8 8428$_H$	5	16
CLMA1	CLMA1 Compare Register H	CLMA1CMPH	03FF$_H$	16	FFF8 842C$_H$	5	16
CLMA1	CLMA1 Protection Instruction Register	CLMA1PCMD	00$_H$	8	FFF8 8430$_H$	5	8
CLMA1	CLMA1 Protection Status Register	CLMA1PS	00$_H$	8	FFF8 8434$_H$	5	8
CLMA2	CLMA2 Control Register 0	CLMA2CTL0	00$_H$	8	FFF8 8440$_H$	5	8
CLMA2	CLMA2 compare Register L	CLMA2CMPL	0001$_H$	16	FFF8 8448$_H$	5	16

(129/201)

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
CLMA2	CLMA2 compare Register H	CLMA2CMPH	$03FF_H$	16	FFF8 $844C_H$	5	16
CLMA2	CLMA2 Protection instruction Register	CLMA2PCMD	00_H	8	FFF8 8450_H	5	8
CLMA2	CLMA2 Protection Status Register	CLMA2PS	00_H	8	FFF8 8454_H	5	8
SYS	Clock 0 Division Register	CLKD0DIV	0000 0004_H	32	FFF8 8800_H	5	32
SYS	Clock 0 Division Status Register	CLKD0STAT	0000 0001_H	32	FFF8 8804_H	5	32
SYS	Clock 0 Selection Control Register	CKSC0CTL	0000 0020_H	32	FFF8 9000_H	5	32
SYS	Clock 0 Selection Active Register	CKSC0ACT	0000 0020_H	32	FFF8 9008_H	5	32
SYS	Clock 1 Selection Control Register	CKSC1CTL	0000 0021_H	32	FFF8 9040_H	5	32
SYS	Clock 1 Selection Active Register	CKSC1ACT	0000 0021_H	32	FFF8 9048_H	5	32
FLASH	FHVE15 Control Register	FHVE15	0000 0000_H	32	FFF8 $A430_H$	5	32
SYS	Power-on Clear Flag Register	POF	0000 $000X_H$	32	FFF8 $AC10_H$	5	32
SYS	Power-on Clear Flag Clear Register	POFC	0000 0000_H	32	FFF8 $AC14_H$	5	32
SYS	Software Reset Request Register	SWRESA	0000 0000_H	32	FFF8 $AC18_H$	5	32
SYS	Protection 1 Command Register	PROT1PHCMD	0000 0000_H	32	FFF8 $B000_H$	5	32
SYS	Protection 1 Status Register	PROT1PS	0000 0000_H	32	FFF8 $B004_H$	5	32
PBG	PBG2A Protection Register 0	FSGD2ADPROT0	07FF $FFFF_H$	32	FFF9 4000_H	2	8, 16, 32
PBG	PBG2A Protection Register 1	FSGD2ADPROT1	07FF $FFFF_H$	32	FFF9 4004_H	2	8, 16, 32
PBG	PBG2A Protection Register 2	FSGD2ADPROT2	07FF $FFFF_H$	32	FFF9 4008_H	2	8, 16, 32
PBG	PBG2A Protection Register 3	FSGD2ADPROT3	07FF $FFFF_H$	32	FFF9 $400C_H$	2	8, 16, 32
PBG	PBG2A Protection Register 4	FSGD2ADPROT4	07FF $FFFF_H$	32	FFF9 4010_H	2	8, 16, 32
PBG	PBG2A Protection Register 5	FSGD2ADPROT5	07FF $FFFF_H$	32	FFF9 4014_H	2	8, 16, 32
PBG	PBG2A Protection Register 6	FSGD2ADPROT6	07FF $FFFF_H$	32	FFF9 4018_H	2	8, 16, 32
PBG	PBG2A Protection Register 7	FSGD2ADPROT7	07FF $FFFF_H$	32	FFF9 $401C_H$	2	8, 16, 32
PBG	PBG2A Error Control Register	ERRSLV2ACTL	0000 0000_H	32	FFF9 4200_H	2	8, 16, 32
PBG	PBG2A Error Status Register	ERRSLV2ASTAT	0000 0000_H	32	FFF9 4204_H	2	8, 16, 32
PBG	PBG2A Error Address Register	ERRSLV2AADDR	0000 0000_H	32	FFF9 4208_H	2	32
PBG	PBG2A Error Type Register	ERRSLV2ATYPE	0000 0000_H	32	FFF9 $420C_H$	2	16, 32
APDP	P-Bus Data Parity Status Register CS0A	APDPERRST_CS0A	0000 0000_H	32	FFF9 6000_H	2	8, 16, 32
APDP	P-Bus Data Parity Status Clear Register CS0A	APDPERRSTC_CS0A	0000 0000_H	32	FFF9 6004_H	2	8, 16, 32
APDP	P-Bus Data Parity Test Mode Control Register CS0A	APDPTMC_CS0A	0000 0000_H	32	FFF9 6008_H	2	16, 32
APDP	P-Bus Data Parity Error Address Register CS0A	APDPERRADR_CS0A	0000 0000_H	32	FFF9 $600C_H$	2	32
APDP	P-Bus Data Parity Status Register CS0B	APDPERRST_CS0B	0000 0000_H	32	FFF9 6020_H	2	8, 16, 32
APDP	P-Bus Data Parity Status Clear Register CS0B	APDPERRSTC_CS0B	0000 0000_H	32	FFF9 6024_H	2	8, 16, 32
APDP	P-Bus Data Parity Test Mode Control Register CS0B	APDPTMC_CS0B	0000 0000_H	32	FFF9 6028_H	2	16, 32
APDP	P-Bus Data Parity Error Address Register CS0B	APDPERRADR_CS0B	0000 0000_H	32	FFF9 $602C_H$	2	32
APDP	P-Bus Data Parity Status Register CS1A	APDPERRST_CS1A	0000 0000_H	32	FFF9 6040_H	2	8, 16, 32
APDP	P-Bus Data Parity Status Clear Register CS1A	APDPERRSTC_CS1A	0000 0000_H	32	FFF9 6044_H	2	8, 16, 32
APDP	P-Bus Data Parity Test Mode Control Register CS1A	APDPTMC_CS1A	0000 0000_H	32	FFF9 6048_H	2	16, 32
APDP	P-Bus Data Parity Error Address Register CS1A	APDPERRADR_CS1A	0000 0000_H	32	FFF9 $604C_H$	2	32
APDP	P-Bus Data Parity Status Register CS1B	APDPERRST_CS1B	0000 0000_H	32	FFF9 6060_H	2	8, 16, 32
APDP	P-Bus Data Parity Status Clear Register CS1B	APDPERRSTC_CS1B	0000 0000_H	32	FFF9 6064_H	2	8, 16, 32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
APDP	P-Bus Data Parity Test Mode Control Register CS1B	APDPTMC_CS1B	0000 0000$_H$	32	FFF9 6068$_H$	2	16, 32
APDP	P-Bus Data Parity Error Address Register CS1B	APDPERRADR_CS1B	0000 0000$_H$	32	FFF9 606C$_H$	2	32
APDP	P-Bus Data Parity Status Register CS2A	APDPERRST_CS2A	0000 0000$_H$	32	FFF9 6080$_H$	2	8, 16, 32
APDP	P-Bus Data Parity Status Clear Register CS2A	APDPERRSTC_CS2A	0000 0000$_H$	32	FFF9 6084$_H$	2	8, 16, 32
APDP	P-Bus Data Parity Test Mode Control Register CS2A	APDPTMC_CS2A	0000 0000$_H$	32	FFF9 6088$_H$	2	16, 32
APDP	P-Bus Data Parity Error Address Register CS2A	APDPERRADR_CS2A	0000 0000$_H$	32	FFF9 608C$_H$	2	32
APDP	P-Bus Data Parity Status Register CS2B	APDPERRST_CS2B	0000 0000$_H$	32	FFF9 60A0$_H$	2	8, 16, 32
APDP	P-Bus Data Parity Status Clear Register CS2B	APDPERRSTC_CS2B	0000 0000$_H$	32	FFF9 60A4$_H$	2	8, 16, 32
APDP	P-Bus Data Parity Test Mode Control Register CS2B	APDPTMC_CS2B	0000 0000$_H$	32	FFF9 60A8$_H$	2	16, 32
APDP	P-Bus Data Parity Error Address Register CS2B	APDPERRADR_CS2B	0000 0000$_H$	32	FFF9 60AC$_H$	2	32
APDP	P-Bus Data Parity Status Register CS3A	APDPERRST_CS3A	0000 0000$_H$	32	FFF9 60C0$_H$	2	8, 16, 32
APDP	P-Bus Data Parity Status Clear Register CS3A	APDPERRSTC_CS3A	0000 0000$_H$	32	FFF9 60C4$_H$	2	8, 16, 32
APDP	P-Bus Data Parity Test Mode Control Register CS3A	APDPTMC_CS3A	0000 0000$_H$	32	FFF9 60C8$_H$	2	16, 32
APDP	P-Bus Data Parity Error Address Register CS3A	APDPERRADR_CS3A	0000 0000$_H$	32	FFF9 60CC$_H$	2	32
APDP	P-Bus Data Parity Status Register CS3B	APDPERRST_CS3B	0000 0000$_H$	32	FFF9 60E0$_H$	2	8, 16, 32
APDP	P-Bus Data Parity Status Clear Register CS3B	APDPERRSTC_CS3B	0000 0000$_H$	32	FFF9 60E4$_H$	2	8, 16, 32
APDP	P-Bus Data Parity Test Mode Control Register CS3B	APDPTMC_CS3B	0000 0000$_H$	32	FFF9 60E8$_H$	2	16, 32
APDP	P-Bus Data Parity Error Address Register CS3B	APDPERRADR_CS3B	0000 0000$_H$	32	FFF9 60EC$_H$	2	32
INTIF	Peripheral Interrupt Status Register 0	PINT0	0000 0000$_H$	32	FFF9 8000$_H$	2	32
INTIF	Peripheral Interrupt Status Register 1	PINT1	0000 0000$_H$	32	FFF9 8004$_H$	2	32
INTIF	Peripheral Interrupt Status Register 2	PINT2	0000 0000$_H$	32	FFF9 8008$_H$	2	32
INTIF	Peripheral Interrupt Status Register 3	PINT3	0000 0000$_H$	32	FFF9 800C$_H$	2	32
INTIF	Peripheral Interrupt Status Register 4	PINT4	0000 0000$_H$	32	FFF9 8010$_H$	2	32
INTIF	Peripheral Interrupt Status Register 5	PINT5	0000 0000$_H$	32	FFF9 8014$_H$	2	32
INTIF	Peripheral Interrupt Status Register 6	PINT6	0000 0000$_H$	32	FFF9 8018$_H$	2	32
INTIF	Peripheral Interrupt Status Register 7	PINT7	0000 0000$_H$	32	FFF9 801C$_H$	2	32
INTIF	Peripheral Interrupt Status Clear Register 0	PINTCLR0	0000 0000$_H$	32	FFF9 8020$_H$	2	32
INTIF	Peripheral Interrupt Status Clear Register 1	PINTCLR1	0000 0000$_H$	32	FFF9 8024$_H$	2	32
INTIF	Peripheral Interrupt Status Clear Register 2	PINTCLR2	0000 0000$_H$	32	FFF9 8028$_H$	2	32
INTIF	Peripheral Interrupt Status Clear Register 3	PINTCLR3	0000 0000$_H$	32	FFF9 802C$_H$	2	32
INTIF	Peripheral Interrupt Status Clear Register 4	PINTCLR4	0000 0000$_H$	32	FFF9 8030$_H$	2	32
INTIF	Peripheral Interrupt Status Clear Register 5	PINTCLR5	0000 0000$_H$	32	FFF9 8034$_H$	2	32
INTIF	Peripheral Interrupt Status Clear Register 6	PINTCLR6	0000 0000$_H$	32	FFF9 8038$_H$	2	32
INTIF	Peripheral Interrupt Status Clear Register 7	PINTCLR7	0000 0000$_H$	32	FFF9 803C$_H$	2	32

(131/201)

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
INTG	Timer Interrupt Mask Enable Register	TIMER	00_H	8	FFF9 8040_H	2	8
IPG	Peripheral Device Protection Violation Access Information Register	IPGECRUM	Undefined (Retention)	16	FFFE $E002_H$	CPU	16
IPG	Peripheral Device Protection Violation Access Address Register	IPGADRUM	Undefined (Retention)	32	FFFE $E008_H$	CPU	32
IPG	Peripheral Device Protection Enable Register	IPGENUM	00_H	8	FFFE $E00D_H$	CPU	8
IPG	Peripheral Device Protection Setting Register 0	IPGPMTUM0	00_H	8	FFFE $E020_H$	CPU	8
IPG	Peripheral Device Protection Setting Register 1	IPGPMTUM1	00_H	8	FFFE $E021_H$	CPU	8
IPG	Peripheral Device Protection Setting Register 2	IPGPMTUM2	00_H	8	FFFE $E022_H$	CPU	8
IPG	Peripheral Device Protection Setting Register 3	IPGPMTUM3	00_H	8	FFFE $E023_H$	CPU	8
IPG	Peripheral Device Protection Setting Register 4	IPGPMTUM4	00_H	8	FFFE $E024_H$	CPU	8
PEG	PE Guard SPID Master Judgement Control Register	PEGSP	0000_H	16	FFFE $E60C_H$	CPU	8, 16
PEG	PE Guard Area 0 mask Setting Register	PEGG0MK	0000 0000_H	32	FFFE $E680_H$	CPU	8, 16, 32
PEG	PE Guard Area 0 Base Setting Register	PEGG0BA	0000 0000_H	32	FFFE $E684_H$	CPU	8, 16, 32
PEG	PE Guard Area 1 mask Setting Register	PEGG1MK	0000 0000_H	32	FFFE $E690_H$	CPU	8, 16, 32
PEG	PE Guard Area 1 Base Setting Register	PEGG1BA	0000 0000_H	32	FFFE $E694_H$	CPU	8, 16, 32
PEG	PE Guard Area 2 Mask Setting Register	PEGG2MK	0000 0000_H	32	FFFE $E6A0_H$	CPU	8, 16, 32
PEG	PE Guard Area 2 Base Setting Register	PEGG2BA	0000 0000_H	32	FFFE $E6A4_H$	CPU	8, 16, 32
PEG	PE Guard Area 3 Mask Setting Register	PEGG3MK	0000 0000_H	32	FFFE $E6B0_H$	CPU	8, 16, 32
PEG	PE Guard Area 3 Base Setting Register	PEGG3BA	0000 0000_H	32	FFFE $E6B4_H$	CPU	8, 16, 32
SEG	Error Notification Control Register	SEGCONT	$C774_H$	16	FFFE $E980_H$	CPU	16
SEG	Error Occurrence Retention Register	SEGFLAG	0000_H	16	FFFE $E982_H$	CPU	16
SEG	Error Source Retention Register (Address)	SEGADDR	Undefined (Retention)	32	FFFE $E988_H$	CPU	16, 32
PCU	System Error Control Register	SEG_CONT	0000_H	16	FFFE $E9F0_H$	CPU	8, 16
PCU	System Error Flag Register	SEG_FLAG	0000_H	16	FFFE $E9F2_H$	CPU	8, 16
INTC1	EI Level Interrupt Control Register 0	EIC0	$008F_H/0087_H$	16	FFFE $EA00_H$	CPU	8, 16
INTC1	EI Level Interrupt Control Register 1	EIC1	$008F_H/0087_H$	16	FFFE $EA02_H$	CPU	8, 16
INTC1	EI Level Interrupt Control Register 2	EIC2	$008F_H/0087_H$	16	FFFE $EA04_H$	CPU	8, 16
INTC1	EI Level Interrupt Control Register 3	EIC3	$008F_H/0087_H$	16	FFFE $EA06_H$	CPU	8, 16
INTC1	EI Level Interrupt Control Register 8	EIC8	$008F_H/0087_H$	16	FFFE $EA10_H$	CPU	8, 16
INTC1	EI Level Interrupt Control Register 9	EIC9	$808F_H/8087_H$	16	FFFE $EA12_H$	CPU	8, 16
INTC1	EI Level Interrupt Control Register 10	EIC10	$808F_H/8087_H$	16	FFFE $EA14_H$	CPU	8, 16
INTC1	EI Level Interrupt Control Register 11	EIC11	$808F_H/8087_H$	16	FFFE $EA16_H$	CPU	8, 16
INTC1	EI Level Interrupt Control Register 12	EIC12	$808F_H/8087_H$	16	FFFE $EA18_H$	CPU	8, 16
INTC1	EI Level Interrupt Control Register 13	EIC13	$808F_H/8087_H$	16	FFFE $EA1A_H$	CPU	8, 16
INTC1	EI Level Interrupt Control Register 14	EIC14	$808F_H/8087_H$	16	FFFE $EA1C_H$	CPU	8, 16
INTC1	EI Level Interrupt Control Register 15	EIC15	$808F_H/8087_H$	16	FFFE $EA1E_H$	CPU	8, 16
INTC1	EI Level Interrupt Control Register 16	EIC16	$808F_H/8087_H$	16	FFFE $EA20_H$	CPU	8, 16
INTC1	EI Level Interrupt Control Register 17	EIC17	$808F_H/8087_H$	16	FFFE $EA22_H$	CPU	8, 16
INTC1	EI Level Interrupt Control Register 18	EIC18	$808F_H/8087_H$	16	FFFE $EA24_H$	CPU	8, 16
INTC1	EI Level Interrupt Control Register 19	EIC19	$808F_H/8087_H$	16	FFFE $EA26_H$	CPU	8, 16
INTC1	EI Level Interrupt Control Register 20	EIC20	$808F_H/8087_H$	16	FFFE $EA28_H$	CPU	8, 16
INTC1	EI Level Interrupt Control Register 25	EIC25	$008F_H/0087_H$	16	FFFE $EA32_H$	CPU	8, 16

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
INTC1	EI Level Interrupt Control Register 26	EIC26	$008F_H/0087_H$	16	FFFE $EA34_H$	CPU	8, 16
INTC1	EI Level Interrupt Control Register 27	EIC27	$008F_H/0087_H$	16	FFFE $EA36_H$	CPU	8, 16
INTC1	EI Level Interrupt Control Register 29	EIC29	$008F_H/0087_H$	16	FFFE $EA3A_H$	CPU	8, 16
INTC1	EI Level Interrupt Mask Register 0	IMR0	$FFFF\ FFFF_H$	32	FFFE $EAF0_H$	CPU	8, 16, 32
INTC1	EI Level Interrupt Bind Register 0	EIBD0	Same value as PEID bit	32	FFFE $EB00_H$	CPU	32
INTC1	EI Level Interrupt Bind Register 1	EIBD1	Same value as PEID bit	32	FFFE $EB04_H$	CPU	32
INTC1	EI Level Interrupt Bind Register 2	EIBD2	Same value as PEID bit	32	FFFE $EB08_H$	CPU	32
INTC1	EI Level Interrupt Bind Register 3	EIBD3	Same value as PEID bit	32	FFFE $EB0C_H$	CPU	32
INTC1	EI Level Interrupt Bind Register 8	EIBD8	Same value as PEID bit	32	FFFE $EB20_H$	CPU	32
INTC1	EI Level Interrupt Bind Register 9	EIBD9	Same value as PEID bit	32	FFFE $EB24_H$	CPU	32
INTC1	EI Level Interrupt Bind Register 10	EIBD10	Same value as PEID bit	32	FFFE $EB28_H$	CPU	32
INTC1	EI Level Interrupt Bind Register 11	EIBD11	Same value as PEID bit	32	FFFE $EB2C_H$	CPU	32
INTC1	EI Level Interrupt Bind Register 12	EIBD12	Same value as PEID bit	32	FFFE $EB30_H$	CPU	32
INTC1	EI Level Interrupt Bind Register 13	EIBD13	Same value as PEID bit	32	FFFE $EB34_H$	CPU	32
INTC1	EI Level Interrupt Bind Register 14	EIBD14	Same value as PEID bit	32	FFFE $EB38_H$	CPU	32
INTC1	EI Level Interrupt Bind Register 15	EIBD15	Same value as PEID bit	32	FFFE $EB3C_H$	CPU	32
INTC1	EI Level Interrupt Bind Register 16	EIBD16	Same value as PEID bit	32	FFFE $EB40_H$	CPU	32
INTC1	EI Level Interrupt Bind Register 17	EIBD17	Same value as PEID bit	32	FFFE $EB44_H$	CPU	32
INTC1	EI Level Interrupt Bind Register 18	EIBD18	Same value as PEID bit	32	FFFE $EB48_H$	CPU	32
INTC1	EI Level Interrupt Bind Register 19	EIBD19	Same value as PEID bit	32	FFFE $EB4C_H$	CPU	32
INTC1	EI Level Interrupt Bind Register 20	EIBD20	Same value as PEID bit	32	FFFE $EB50_H$	CPU	32
INTC1	EI Level Interrupt Bind Register 25	EIBD25	Same value as PEID bit	32	FFFE $EB64_H$	CPU	32
INTC1	EI Level Interrupt Bind Register 26	EIBD26	Same value as PEID bit	32	FFFE $EB68_H$	CPU	32
INTC1	EI Level Interrupt Bind Register 27	EIBD27	Same value as PEID bit	32	FFFE $EB6C_H$	CPU	32
INTC1	EI Level Interrupt Bind Register 29	EIBD29	Same value as PEID bit	32	FFFE $EB74_H$	CPU	32
MEV	Exclusive Control Register 0	G0MEV0	$0000\ 0000_H$	32	FFFE $EC00_H$	CPU	8, 16, 32
MEV	Exclusive Control Register 1	G0MEV1	$0000\ 0000_H$	32	FFFE $EC04_H$	CPU	8, 16, 32
MEV	Exclusive Control Register 2	G0MEV2	$0000\ 0000_H$	32	FFFE $EC08_H$	CPU	8, 16, 32
MEV	Exclusive Control Register 3	G0MEV3	$0000\ 0000_H$	32	FFFE $EC0C_H$	CPU	8, 16, 32
MEV	Exclusive Control Register 4	G0MEV4	$0000\ 0000_H$	32	FFFE $EC10_H$	CPU	8, 16, 32
MEV	Exclusive Control Register 5	G0MEV5	$0000\ 0000_H$	32	FFFE $EC14_H$	CPU	8, 16, 32
MEV	Exclusive Control Register 6	G0MEV6	$0000\ 0000_H$	32	FFFE $EC18_H$	CPU	8, 16, 32
MEV	Exclusive Control Register 7	G0MEV7	$0000\ 0000_H$	32	FFFE $EC1C_H$	CPU	8, 16, 32
MEV	Exclusive Control Register 8	G0MEV8	$0000\ 0000_H$	32	FFFE $EC20_H$	CPU	8, 16, 32
MEV	Exclusive Control Register 9	G0MEV9	$0000\ 0000_H$	32	FFFE $EC24_H$	CPU	8, 16, 32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
MEV	Exclusive Control Register 10	G0MEV10	0000 0000$_H$	32	FFFE EC28$_H$	CPU	8, 16, 32
MEV	Exclusive Control Register 11	G0MEV11	0000 0000$_H$	32	FFFE EC2C$_H$	CPU	8, 16, 32
MEV	Exclusive Control Register 12	G0MEV12	0000 0000$_H$	32	FFFE EC30$_H$	CPU	8, 16, 32
MEV	Exclusive Control Register 13	G0MEV13	0000 0000$_H$	32	FFFE EC34$_H$	CPU	8, 16, 32
MEV	Exclusive Control Register 14	G0MEV14	0000 0000$_H$	32	FFFE EC38$_H$	CPU	8, 16, 32
MEV	Exclusive Control Register 15	G0MEV15	0000 0000$_H$	32	FFFE EC3C$_H$	CPU	8, 16, 32
MEV	Exclusive Control Register 16	G0MEV16	0000 0000$_H$	32	FFFE EC40$_H$	CPU	8, 16, 32
MEV	Exclusive Control Register 17	G0MEV17	0000 0000$_H$	32	FFFE EC44$_H$	CPU	8, 16, 32
MEV	Exclusive Control Register 18	G0MEV18	0000 0000$_H$	32	FFFE EC48$_H$	CPU	8, 16, 32
MEV	Exclusive Control Register 19	G0MEV19	0000 0000$_H$	32	FFFE EC4C$_H$	CPU	8, 16, 32
MEV	Exclusive Control Register 20	G0MEV20	0000 0000$_H$	32	FFFE EC50$_H$	CPU	8, 16, 32
MEV	Exclusive Control Register 21	G0MEV21	0000 0000$_H$	32	FFFE EC54$_H$	CPU	8, 16, 32
MEV	Exclusive Control Register 22	G0MEV22	0000 0000$_H$	32	FFFE EC58$_H$	CPU	8, 16, 32
MEV	Exclusive Control Register 23	G0MEV23	0000 0000$_H$	32	FFFE EC5C$_H$	CPU	8, 16, 32
MEV	Exclusive Control Register 24	G0MEV24	0000 0000$_H$	32	FFFE EC60$_H$	CPU	8, 16, 32
MEV	Exclusive Control Register 25	G0MEV25	0000 0000$_H$	32	FFFE EC64$_H$	CPU	8, 16, 32
MEV	Exclusive Control Register 26	G0MEV26	0000 0000$_H$	32	FFFE EC68$_H$	CPU	8, 16, 32
MEV	Exclusive Control Register 27	G0MEV27	0000 0000$_H$	32	FFFE EC6C$_H$	CPU	8, 16, 32
MEV	Exclusive Control Register 28	G0MEV28	0000 0000$_H$	32	FFFE EC70$_H$	CPU	8, 16, 32
MEV	Exclusive Control Register 29	G0MEV29	0000 0000$_H$	32	FFFE EC74$_H$	CPU	8, 16, 32
MEV	Exclusive Control Register 30	G0MEV30	0000 0000$_H$	32	FFFE EC78$_H$	CPU	8, 16, 32
MEV	Exclusive Control Register 31	G0MEV31	0000 0000$_H$	32	FFFE EC7C$_H$	CPU	8, 16, 32
IPIRSS	Inter-PE Interrupt Register 0	IPIR_CH0	0000 0000$_H$	32	FFFE EC80$_H$	CPU	8, 16, 32
IPIRSS	Inter-PE Interrupt Register 1	IPIR_CH1	0000 0000$_H$	32	FFFE EC84$_H$	CPU	8, 16, 32
IPIRSS	Inter-PE Interrupt Register 2	IPIR_CH2	0000 0000$_H$	32	FFFE EC88$_H$	CPU	8, 16, 32
IPIRSS	Inter-PE Interrupt Register 3	IPIR_CH3	0000 0000$_H$	32	FFFE EC8C$_H$	CPU	8, 16, 32
TESTCOMP	Comparator Test Register 0	TESTCOMPREG0	0000 0000$_H$	32	FFFE ED00$_H$	CPU	8, 16, 32
TESTCOMP	Comparator Test Register 1	TESTCOMPREG1	0000 0000$_H$	32	FFFE ED04$_H$	CPU	8, 16, 32
CFU	Cache Clear Operation Register	TM_CC	0000 0000$_H$	32	FFFF 7808$_H$	0	32
CFU	ERAM Bank 0 Mapping Enable Register	TM_ME	0000 0000$_H$	32	FFFF 7810$_H$	0	32
CFU	ERAM Bank 0 Mapping Status Register	TM_MS	0000 0000$_H$	32	FFFF 7814$_H$	0	32
CFU	ERAM Bank 0 Mapping Size Setting Register	TM_BMC0	0000 0000$_H$	32	FFFF 7830$_H$	0	32
CFU	ERAM Bank 0 Mapping Address Register	TM_MA0	0000 0000$_H$	32	FFFF 7840$_H$	0	32
DMASS	DMA Control Register	DMACTL	0000 0000$_H$	32	FFFF 8000$_H$	0	32
DMASS	DTS Control Register 1	DTSCTL1	0000 0000$_H$	32	FFFF 8010$_H$	0	32
DMASS	DTS Control Register 2	DTSCTL2	0000 0000$_H$	32	FFFF 8014$_H$	0	32
DMASS	DTS State Register	DTSSTS	0000 0000$_H$	32	FFFF 8018$_H$	0	32
DMASS	DMAC Error Register	DMACER	0000 0000$_H$	32	FFFF 8020$_H$	0	32
DMASS	DTS Error Register 1	DTSER1	0000 0000$_H$	32	FFFF 8024$_H$	0	32
DMASS	DTS Error Register 2	DTSER2	0000 0000$_H$	32	FFFF 8028$_H$	0	32
DMASS	DTS Error Clear Register	DTSERC	0000 0000$_H$	32	FFFF 802C$_H$	0	32
DMASS	DMAC0 Register Access Protection Violation Register	DM0CMV	0000 0000$_H$	32	FFFF 8030$_H$	0	32
DMASS	DMAC1 Register Access Protection Violation Register	DM1CMV	0000 0000$_H$	32	FFFF 8034$_H$	0	32
DMASS	DTS Register Access Protection Violation Register	DTSCMV	0000 0000$_H$	32	FFFF 8038$_H$	0	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
DMASS	Register Access Protection Violation Clear Register	CMVC	0000 0000$_H$	32	FFFF 803C$_H$	0	32
DMASS	DTS Channel Priority Setting Register 0	DTSPR0	0000 0000$_H$	32	FFFF 8060$_H$	0	32
DMASS	DTS Channel Priority Setting Register 1	DTSPR1	0000 0000$_H$	32	FFFF 8064$_H$	0	32
DMASS	DTS Channel Priority Setting Register 2	DTSPR2	0000 0000$_H$	32	FFFF 8068$_H$	0	32
DMASS	DTS Channel Priority Setting Register 3	DTSPR3	0000 0000$_H$	32	FFFF 806C$_H$	0	32
DMASS	DTS Channel Priority Setting Register 4	DTSPR4	0000 0000$_H$	32	FFFF 8070$_H$	0	32
DMASS	DTS Channel Priority Setting Register 5	DTSPR5	0000 0000$_H$	32	FFFF 8074$_H$	0	32
DMASS	DTS Channel Priority Setting Register 6	DTSPR6	0000 0000$_H$	32	FFFF 8078$_H$	0	32
DMASS	DTS Channel Priority Setting Register 7	DTSPR7	0000 0000$_H$	32	FFFF 807C$_H$	0	32
DMASS	DTSRAM ECC Control Register	DTRECCTL	0000 0000$_H$	32	FFFF 8080$_H$	0	32
DMASS	DTSRAM Error Notification Control Register	DTRERINT	0000 0002$_H$	32	FFFF 8084$_H$	0	32
DMASS	DTSRAM Test Control Register	DTRTSCTL	0000 0000$_H$	32	FFFF 8094$_H$	0	32
DMASS	DTSRAM Test Write Data Register	DTRTWDAT	0000 0000$_H$	32	FFFF 8098$_H$	0	32
DMASS	DTSRAM Test Read Data Register	DTRTRDAT	0000 0000$_H$	32	FFFF 809C$_H$	0	32
DMASS	DMAC0 Channel 0 Channel Master Setting Register	DM00CM	0000 0010$_H$	32	FFFF 8100$_H$	0	32
DMASS	DMAC0 Channel 1 Channel Master Setting Register	DM01CM	0000 0010$_H$	32	FFFF 8104$_H$	0	32
DMASS	DMAC0 Channel 2 Channel Master Setting Register	DM02CM	0000 0010$_H$	32	FFFF 8108$_H$	0	32
DMASS	DMAC0 Channel 3 Channel Master Setting Register	DM03CM	0000 0010$_H$	32	FFFF 810C$_H$	0	32
DMASS	DMAC0 Channel 4 Channel Master Setting Register	DM04CM	0000 0010$_H$	32	FFFF 8110$_H$	0	32
DMASS	DMAC0 Channel 5 Channel Master Setting Register	DM05CM	0000 0010$_H$	32	FFFF 8114$_H$	0	32
DMASS	DMAC0 Channel 6 Channel Master Setting Register	DM06CM	0000 0010$_H$	32	FFFF 8118$_H$	0	32
DMASS	DMAC0 Channel 7 Channel Master Setting Register	DM07CM	0000 0010$_H$	32	FFFF 811C$_H$	0	32
DMASS	DMAC1 Channel 0 Channel Master Setting Register	DM10CM	0000 0010$_H$	32	FFFF 8120$_H$	0	32
DMASS	DMAC1 Channel 1 Channel Master Setting Register	DM11CM	0000 0010$_H$	32	FFFF 8124$_H$	0	32
DMASS	DMAC1 Channel 2 Channel Master Setting Register	DM12CM	0000 0010$_H$	32	FFFF 8128$_H$	0	32
DMASS	DMAC1 Channel 3 Channel Master Setting Register	DM13CM	0000 0010$_H$	32	FFFF 812C$_H$	0	32
DMASS	DMAC1 Channel 4 Channel Master Setting Register	DM14CM	0000 0010$_H$	32	FFFF 8130$_H$	0	32
DMASS	DMAC1 Channel 5 Channel Master Setting Register	DM15CM	0000 0010$_H$	32	FFFF 8134$_H$	0	32
DMASS	DMAC1 Channel 6 Channel Master Setting Register	DM16CM	0000 0010$_H$	32	FFFF 8138$_H$	0	32
DMASS	DMAC1 Channel 7 Channel Master Setting Register	DM17CM	0000 0010$_H$	32	FFFF 813C$_H$	0	32
DMASS	DTS Channel 000 Channel Master Setting Register	DTS000CM	XXXX XXXX$_H$	32	FFFF 8200$_H$	0	32
DMASS	DTS Channel 001 Channel Master Setting Register	DTS001CM	XXXX XXXX$_H$	32	FFFF 8204$_H$	0	32
DMASS	DTS Channel 002 Channel Master Setting Register	DTS002CM	XXXX XXXX$_H$	32	FFFF 8208$_H$	0	32
DMASS	DTS Channel 003 Channel Master Setting Register	DTS003CM	XXXX XXXX$_H$	32	FFFF 820C$_H$	0	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
DMASS	DTS Channel 004 Channel Master Setting Register	DTS004CM	XXXX $XXXX_H$	32	FFFF 8210_H	0	32
DMASS	DTS Channel 005 Channel Master Setting Register	DTS005CM	XXXX $XXXX_H$	32	FFFF 8214_H	0	32
DMASS	DTS Channel 006 Channel Master Setting Register	DTS006CM	XXXX $XXXX_H$	32	FFFF 8218_H	0	32
DMASS	DTS Channel 007 Channel Master Setting Register	DTS007CM	XXXX $XXXX_H$	32	FFFF $821C_H$	0	32
DMASS	DTS Channel 008 Channel Master Setting Register	DTS008CM	XXXX $XXXX_H$	32	FFFF 8220_H	0	32
DMASS	DTS Channel 009 Channel Master Setting Register	DTS009CM	XXXX $XXXX_H$	32	FFFF 8224_H	0	32
DMASS	DTS Channel 010 Channel Master Setting Register	DTS010CM	XXXX $XXXX_H$	32	FFFF 8228_H	0	32
DMASS	DTS Channel 011 Channel Master Setting Register	DTS011CM	XXXX $XXXX_H$	32	FFFF $822C_H$	0	32
DMASS	DTS Channel 012 Channel Master Setting Register	DTS012CM	XXXX $XXXX_H$	32	FFFF 8230_H	0	32
DMASS	DTS Channel 013 Channel Master Setting Register	DTS013CM	XXXX $XXXX_H$	32	FFFF 8234_H	0	32
DMASS	DTS Channel 014 Channel Master Setting Register	DTS014CM	XXXX $XXXX_H$	32	FFFF 8238_H	0	32
DMASS	DTS Channel 015 Channel Master Setting Register	DTS015CM	XXXX $XXXX_H$	32	FFFF $823C_H$	0	32
DMASS	DTS Channel 016 Channel Master Setting Register	DTS016CM	XXXX $XXXX_H$	32	FFFF 8240_H	0	32
DMASS	DTS Channel 017 Channel Master Setting Register	DTS017CM	XXXX $XXXX_H$	32	FFFF 8244_H	0	32
DMASS	DTS Channel 018 Channel Master Setting Register	DTS018CM	XXXX $XXXX_H$	32	FFFF 8248_H	0	32
DMASS	DTS Channel 019 Channel Master Setting Register	DTS019CM	XXXX $XXXX_H$	32	FFFF $824C_H$	0	32
DMASS	DTS Channel 020 Channel Master Setting Register	DTS020CM	XXXX $XXXX_H$	32	FFFF 8250_H	0	32
DMASS	DTS Channel 021 Channel Master Setting Register	DTS021CM	XXXX $XXXX_H$	32	FFFF 8254_H	0	32
DMASS	DTS Channel 022 Channel Master Setting Register	DTS022CM	XXXX $XXXX_H$	32	FFFF 8258_H	0	32
DMASS	DTS Channel 023 Channel Master Setting Register	DTS023CM	XXXX $XXXX_H$	32	FFFF $825C_H$	0	32
DMASS	DTS Channel 024 Channel Master Setting Register	DTS024CM	XXXX $XXXX_H$	32	FFFF 8260_H	0	32
DMASS	DTS Channel 025 Channel Master Setting Register	DTS025CM	XXXX $XXXX_H$	32	FFFF 8264_H	0	32
DMASS	DTS Channel 026 Channel Master Setting Register	DTS026CM	XXXX $XXXX_H$	32	FFFF 8268_H	0	32
DMASS	DTS Channel 027 Channel Master Setting Register	DTS027CM	XXXX $XXXX_H$	32	FFFF $826C_H$	0	32
DMASS	DTS Channel 028 Channel Master Setting Register	DTS028CM	XXXX $XXXX_H$	32	FFFF 8270_H	0	32
DMASS	DTS Channel 029 Channel Master Setting Register	DTS029CM	XXXX $XXXX_H$	32	FFFF 8274_H	0	32
DMASS	DTS Channel 030 Channel Master Setting Register	DTS030CM	XXXX $XXXX_H$	32	FFFF 8278_H	0	32
DMASS	DTS Channel 031 Channel Master Setting Register	DTS031CM	XXXX $XXXX_H$	32	FFFF $827C_H$	0	32
DMASS	DTS Channel 032 Channel Master Setting Register	DTS032CM	XXXX $XXXX_H$	32	FFFF 8280_H	0	32
DMASS	DTS Channel 033 Channel Master Setting Register	DTS033CM	XXXX $XXXX_H$	32	FFFF 8284_H	0	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
DMASS	DTS Channel 034 Channel Master Setting Register	DTS034CM	XXXX XXXX$_H$	32	FFFF 8288$_H$	0	32
DMASS	DTS Channel 035 Channel Master Setting Register	DTS035CM	XXXX XXXX$_H$	32	FFFF 828C$_H$	0	32
DMASS	DTS Channel 036 Channel Master Setting Register	DTS036CM	XXXX XXXX$_H$	32	FFFF 8290$_H$	0	32
DMASS	DTS Channel 037 Channel Master Setting Register	DTS037CM	XXXX XXXX$_H$	32	FFFF 8294$_H$	0	32
DMASS	DTS Channel 038 Channel Master Setting Register	DTS038CM	XXXX XXXX$_H$	32	FFFF 8298$_H$	0	32
DMASS	DTS Channel 039 Channel Master Setting Register	DTS039CM	XXXX XXXX$_H$	32	FFFF 829C$_H$	0	32
DMASS	DTS Channel 040 Channel Master Setting Register	DTS040CM	XXXX XXXX$_H$	32	FFFF 82A0$_H$	0	32
DMASS	DTS Channel 041 Channel Master Setting Register	DTS041CM	XXXX XXXX$_H$	32	FFFF 82A4$_H$	0	32
DMASS	DTS Channel 042 Channel Master Setting Register	DTS042CM	XXXX XXXX$_H$	32	FFFF 82A8$_H$	0	32
DMASS	DTS Channel 043 Channel Master Setting Register	DTS043CM	XXXX XXXX$_H$	32	FFFF 82AC$_H$	0	32
DMASS	DTS Channel 044 Channel Master Setting Register	DTS044CM	XXXX XXXX$_H$	32	FFFF 82B0$_H$	0	32
DMASS	DTS Channel 045 Channel Master Setting Register	DTS045CM	XXXX XXXX$_H$	32	FFFF 82B4$_H$	0	32
DMASS	DTS Channel 046 Channel Master Setting Register	DTS046CM	XXXX XXXX$_H$	32	FFFF 82B8$_H$	0	32
DMASS	DTS Channel 047 Channel Master Setting Register	DTS047CM	XXXX XXXX$_H$	32	FFFF 82BC$_H$	0	32
DMASS	DTS Channel 048 Channel Master Setting Register	DTS048CM	XXXX XXXX$_H$	32	FFFF 82C0$_H$	0	32
DMASS	DTS Channel 049 Channel Master Setting Register	DTS049CM	XXXX XXXX$_H$	32	FFFF 82C4$_H$	0	32
DMASS	DTS Channel 050 Channel Master Setting Register	DTS050CM	XXXX XXXX$_H$	32	FFFF 82C8$_H$	0	32
DMASS	DTS Channel 051 Channel Master Setting Register	DTS051CM	XXXX XXXX$_H$	32	FFFF 82CC$_H$	0	32
DMASS	DTS Channel 052 Channel Master Setting Register	DTS052CM	XXXX XXXX$_H$	32	FFFF 82D0$_H$	0	32
DMASS	DTS Channel 053 Channel Master Setting Register	DTS053CM	XXXX XXXX$_H$	32	FFFF 82D4$_H$	0	32
DMASS	DTS Channel 054 Channel Master Setting Register	DTS054CM	XXXX XXXX$_H$	32	FFFF 82D8$_H$	0	32
DMASS	DTS Channel 055 Channel Master Setting Register	DTS055CM	XXXX XXXX$_H$	32	FFFF 82DC$_H$	0	32
DMASS	DTS Channel 056 Channel Master Setting Register	DTS056CM	XXXX XXXX$_H$	32	FFFF 82E0$_H$	0	32
DMASS	DTS Channel 057 Channel Master Setting Register	DTS057CM	XXXX XXXX$_H$	32	FFFF 82E4$_H$	0	32
DMASS	DTS Channel 058 Channel Master Setting Register	DTS058CM	XXXX XXXX$_H$	32	FFFF 82E8$_H$	0	32
DMASS	DTS Channel 059 Channel Master Setting Register	DTS059CM	XXXX XXXX$_H$	32	FFFF 82EC$_H$	0	32
DMASS	DTS Channel 060 Channel Master Setting Register	DTS060CM	XXXX XXXX$_H$	32	FFFF 82F0$_H$	0	32
DMASS	DTS Channel 061 Channel Master Setting Register	DTS061CM	XXXX XXXX$_H$	32	FFFF 82F4$_H$	0	32
DMASS	DTS Channel 062 Channel Master Setting Register	DTS062CM	XXXX XXXX$_H$	32	FFFF 82F8$_H$	0	32
DMASS	DTS Channel 063 Channel Master Setting Register	DTS063CM	XXXX XXXX$_H$	32	FFFF 82FC$_H$	0	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
DMASS	DTS Channel 064 Channel Master Setting Register	DTS064CM	XXXX $XXXX_H$	32	FFFF 8300_H	0	32
DMASS	DTS Channel 065 Channel Master Setting Register	DTS065CM	XXXX $XXXX_H$	32	FFFF 8304_H	0	32
DMASS	DTS Channel 066 Channel Master Setting Register	DTS066CM	XXXX $XXXX_H$	32	FFFF 8308_H	0	32
DMASS	DTS Channel 067 Channel Master Setting Register	DTS067CM	XXXX $XXXX_H$	32	FFFF $830C_H$	0	32
DMASS	DTS Channel 068 Channel Master Setting Register	DTS068CM	XXXX $XXXX_H$	32	FFFF 8310_H	0	32
DMASS	DTS Channel 069 Channel Master Setting Register	DTS069CM	XXXX $XXXX_H$	32	FFFF 8314_H	0	32
DMASS	DTS Channel 070 Channel Master Setting Register	DTS070CM	XXXX $XXXX_H$	32	FFFF 8318_H	0	32
DMASS	DTS Channel 071 Channel Master Setting Register	DTS071CM	XXXX $XXXX_H$	32	FFFF $831C_H$	0	32
DMASS	DTS Channel 072 Channel Master Setting Register	DTS072CM	XXXX $XXXX_H$	32	FFFF 8320_H	0	32
DMASS	DTS Channel 073 Channel Master Setting Register	DTS073CM	XXXX $XXXX_H$	32	FFFF 8324_H	0	32
DMASS	DTS Channel 074 Channel Master Setting Register	DTS074CM	XXXX $XXXX_H$	32	FFFF 8328_H	0	32
DMASS	DTS Channel 075 Channel Master Setting Register	DTS075CM	XXXX $XXXX_H$	32	FFFF $832C_H$	0	32
DMASS	DTS Channel 076 Channel Master Setting Register	DTS076CM	XXXX $XXXX_H$	32	FFFF 8330_H	0	32
DMASS	DTS Channel 077 Channel Master Setting Register	DTS077CM	XXXX $XXXX_H$	32	FFFF 8334_H	0	32
DMASS	DTS Channel 078 Channel Master Setting Register	DTS078CM	XXXX $XXXX_H$	32	FFFF 8338_H	0	32
DMASS	DTS Channel 079 Channel Master Setting Register	DTS079CM	XXXX $XXXX_H$	32	FFFF $833C_H$	0	32
DMASS	DTS Channel 080 Channel Master Setting Register	DTS080CM	XXXX $XXXX_H$	32	FFFF 8340_H	0	32
DMASS	DTS Channel 081 Channel Master Setting Register	DTS081CM	XXXX $XXXX_H$	32	FFFF 8344_H	0	32
DMASS	DTS Channel 082 Channel Master Setting Register	DTS082CM	XXXX $XXXX_H$	32	FFFF 8348_H	0	32
DMASS	DTS Channel 083 Channel Master Setting Register	DTS083CM	XXXX $XXXX_H$	32	FFFF $834C_H$	0	32
DMASS	DTS Channel 084 Channel Master Setting Register	DTS084CM	XXXX $XXXX_H$	32	FFFF 8350_H	0	32
DMASS	DTS Channel 085 Channel Master Setting Register	DTS085CM	XXXX $XXXX_H$	32	FFFF 8354_H	0	32
DMASS	DTS Channel 086 Channel Master Setting Register	DTS086CM	XXXX $XXXX_H$	32	FFFF 8358_H	0	32
DMASS	DTS Channel 087 Channel Master Setting Register	DTS087CM	XXXX $XXXX_H$	32	FFFF $835C_H$	0	32
DMASS	DTS Channel 088 Channel Master Setting Register	DTS088CM	XXXX $XXXX_H$	32	FFFF 8360_H	0	32
DMASS	DTS Channel 089 Channel Master Setting Register	DTS089CM	XXXX $XXXX_H$	32	FFFF 8364_H	0	32
DMASS	DTS Channel 090 Channel Master Setting Register	DTS090CM	XXXX $XXXX_H$	32	FFFF 8368_H	0	32
DMASS	DTS Channel 091 Channel Master Setting Register	DTS091CM	XXXX $XXXX_H$	32	FFFF $836C_H$	0	32
DMASS	DTS Channel 092 Channel Master Setting Register	DTS092CM	XXXX $XXXX_H$	32	FFFF 8370_H	0	32
DMASS	DTS Channel 093 Channel Master Setting Register	DTS093CM	XXXX $XXXX_H$	32	FFFF 8374_H	0	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
DMASS	DTS Channel 094 Channel Master Setting Register	DTS094CM	XXXX $XXXX_H$	32	FFFF 8378_H	0	32
DMASS	DTS Channel 095 Channel Master Setting Register	DTS095CM	XXXX $XXXX_H$	32	FFFF $837C_H$	0	32
DMASS	DTS Channel 096 Channel Master Setting Register	DTS096CM	XXXX $XXXX_H$	32	FFFF 8380_H	0	32
DMASS	DTS Channel 097 Channel Master Setting Register	DTS097CM	XXXX $XXXX_H$	32	FFFF 8384_H	0	32
DMASS	DTS Channel 098 Channel Master Setting Register	DTS098CM	XXXX $XXXX_H$	32	FFFF 8388_H	0	32
DMASS	DTS Channel 099 Channel Master Setting Register	DTS099CM	XXXX $XXXX_H$	32	FFFF $838C_H$	0	32
DMASS	DTS Channel 100 Channel Master Setting Register	DTS100CM	XXXX $XXXX_H$	32	FFFF 8390_H	0	32
DMASS	DTS Channel 101 Channel Master Setting Register	DTS101CM	XXXX $XXXX_H$	32	FFFF 8394_H	0	32
DMASS	DTS Channel 102 Channel Master Setting Register	DTS102CM	XXXX $XXXX_H$	32	FFFF 8398_H	0	32
DMASS	DTS Channel 103 Channel Master Setting Register	DTS103CM	XXXX $XXXX_H$	32	FFFF $839C_H$	0	32
DMASS	DTS Channel 104 Channel Master Setting Register	DTS104CM	XXXX $XXXX_H$	32	FFFF $83A0_H$	0	32
DMASS	DTS Channel 105 Channel Master Setting Register	DTS105CM	XXXX $XXXX_H$	32	FFFF $83A4_H$	0	32
DMASS	DTS Channel 106 Channel Master Setting Register	DTS106CM	XXXX $XXXX_H$	32	FFFF $83A8_H$	0	32
DMASS	DTS Channel 107 Channel Master Setting Register	DTS107CM	XXXX $XXXX_H$	32	FFFF $83AC_H$	0	32
DMASS	DTS Channel 108 Channel Master Setting Register	DTS108CM	XXXX $XXXX_H$	32	FFFF $83B0_H$	0	32
DMASS	DTS Channel 109 Channel Master Setting Register	DTS109CM	XXXX $XXXX_H$	32	FFFF $83B4_H$	0	32
DMASS	DTS Channel 110 Channel Master Setting Register	DTS110CM	XXXX $XXXX_H$	32	FFFF $83B8_H$	0	32
DMASS	DTS Channel 111 Channel Master Setting Register	DTS111CM	XXXX $XXXX_H$	32	FFFF $83BC_H$	0	32
DMASS	DTS Channel 112 Channel Master Setting Register	DTS112CM	XXXX $XXXX_H$	32	FFFF $83C0_H$	0	32
DMASS	DTS Channel 113 Channel Master Setting Register	DTS113CM	XXXX $XXXX_H$	32	FFFF $83C4_H$	0	32
DMASS	DTS Channel 114 Channel Master Setting Register	DTS114CM	XXXX $XXXX_H$	32	FFFF $83C8_H$	0	32
DMASS	DTS Channel 115 Channel Master Setting Register	DTS115CM	XXXX $XXXX_H$	32	FFFF $83CC_H$	0	32
DMASS	DTS Channel 116 Channel Master Setting Register	DTS116CM	XXXX $XXXX_H$	32	FFFF $83D0_H$	0	32
DMASS	DTS Channel 117 Channel Master Setting Register	DTS117CM	XXXX $XXXX_H$	32	FFFF $83D4_H$	0	32
DMASS	DTS Channel 118 Channel Master Setting Register	DTS118CM	XXXX $XXXX_H$	32	FFFF $83D8_H$	0	32
DMASS	DTS Channel 119 Channel Master Setting Register	DTS119CM	XXXX $XXXX_H$	32	FFFF $83DC_H$	0	32
DMASS	DTS Channel 120 Channel Master Setting Register	DTS120CM	XXXX $XXXX_H$	32	FFFF $83E0_H$	0	32
DMASS	DTS Channel 121 Channel Master Setting Register	DTS121CM	XXXX $XXXX_H$	32	FFFF $83E4_H$	0	32
DMASS	DTS Channel 122 Channel Master Setting Register	DTS122CM	XXXX $XXXX_H$	32	FFFF $83E8_H$	0	32
DMASS	DTS Channel 123 Channel Master Setting Register	DTS123CM	XXXX $XXXX_H$	32	FFFF $83EC_H$	0	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
DMASS	DTS Channel 124 Channel Master Setting Register	DTS124CM	XXXX $XXXX_H$	32	FFFF $83F0_H$	0	32
DMASS	DTS Channel 125 Channel Master Setting Register	DTS125CM	XXXX $XXXX_H$	32	FFFF $83F4_H$	0	32
DMASS	DTS Channel 126 Channel Master Setting Register	DTS126CM	XXXX $XXXX_H$	32	FFFF $83F8_H$	0	32
DMASS	DTS Channel 127 Channel Master Setting Register	DTS127CM	XXXX $XXXX_H$	32	FFFF $83FC_H$	0	32
DMASS	DMAC Source Address Register	DSA0	$0000\ 0000_H$	32	FFFF 8400_H	0	32
DMASS	DMAC Destination Address Register	DDA0	$0000\ 0000_H$	32	FFFF 8404_H	0	32
DMASS	DMAC Transfer Count Register	DTC0	$0000\ 0000_H$	32	FFFF 8408_H	0	32
DMASS	DMAC Transfer Control Register	DTCT0	$0000\ 0000_H$	32	FFFF $840C_H$	0	32
DMASS	DMAC Reload Source Address Register	DRSA0	$0000\ 0000_H$	32	FFFF 8410_H	0	32
DMASS	DMAC Reload Destination Address Register	DRDA0	$0000\ 0000_H$	32	FFFF 8414_H	0	32
DMASS	DMAC Reload Transfer Count Register	DRTC0	$0000\ 0000_H$	32	FFFF 8418_H	0	32
DMASS	DMAC Transfer Count Compare Register	DTCC0	$0000\ 0000_H$	32	FFFF $841C_H$	0	32
DMASS	DMAC Channel Operation Enable Setting Register	DCEN0	$0000\ 0000_H$	32	FFFF 8420_H	0	32
DMASS	DMAC Transfer Status Register	DCST0	$0000\ 0000_H$	32	FFFF 8424_H	0	32
DMASS	DMAC Transfer Status Set Register	DCSTS0	$0000\ 0000_H$	32	FFFF 8428_H	0	32
DMASS	DMAC Transfer Status Clear Register	DCSTC0	$0000\ 0000_H$	32	FFFF $842C_H$	0	32
DMASS	DTFR Setting Register	DTFR0	$0000\ 0000_H$	32	FFFF 8430_H	0	32
DMASS	DTFR Transfer Request Status Register	DTFRRQ0	$0000\ 0000_H$	32	FFFF 8434_H	0	32
DMASS	DTFR Transfer Request Clear Register	DTFRRQC0	$0000\ 0000_H$	32	FFFF 8438_H	0	32
DMASS	DMAC Source Address Register	DSA1	$0000\ 0000_H$	32	FFFF 8440_H	0	32
DMASS	DMAC Destination Address Register	DDA1	$0000\ 0000_H$	32	FFFF 8444_H	0	32
DMASS	DMAC Transfer Count Register	DTC1	$0000\ 0000_H$	32	FFFF 8448_H	0	32
DMASS	DMAC Transfer Control Register	DTCT1	$0000\ 0000_H$	32	FFFF $844C_H$	0	32
DMASS	DMAC Reload Source Address Register	DRSA1	$0000\ 0000_H$	32	FFFF 8450_H	0	32
DMASS	DMAC Reload Destination Address Register	DRDA1	$0000\ 0000_H$	32	FFFF 8454_H	0	32
DMASS	DMAC Reload Transfer Count Register	DRTC1	$0000\ 0000_H$	32	FFFF 8458_H	0	32
DMASS	DMAC Transfer Count Compare Register	DTCC1	$0000\ 0000_H$	32	FFFF $845C_H$	0	32
DMASS	DMAC Channel Operation Enable Setting Register	DCEN1	$0000\ 0000_H$	32	FFFF 8460_H	0	32
DMASS	DMAC Transfer Status Register	DCST1	$0000\ 0000_H$	32	FFFF 8464_H	0	32
DMASS	DMAC Transfer Status Set Register	DCSTS1	$0000\ 0000_H$	32	FFFF 8468_H	0	32
DMASS	DMAC Transfer Status Clear Register	DCSTC1	$0000\ 0000_H$	32	FFFF $846C_H$	0	32
DMASS	DTFR Setting Register	DTFR1	$0000\ 0000_H$	32	FFFF 8470_H	0	32
DMASS	DTFR Transfer Request Status Register	DTFRRQ1	$0000\ 0000_H$	32	FFFF 8474_H	0	32
DMASS	DTFR Transfer Request Clear Register	DTFRRQC1	$0000\ 0000_H$	32	FFFF 8478_H	0	32
DMASS	DMAC Source Address Register	DSA2	$0000\ 0000_H$	32	FFFF 8480_H	0	32
DMASS	DMAC Destination Address Register	DDA2	$0000\ 0000_H$	32	FFFF 8484_H	0	32
DMASS	DMAC Transfer Count Register	DTC2	$0000\ 0000_H$	32	FFFF 8488_H	0	32
DMASS	DMAC Transfer Control Register	DTCT2	$0000\ 0000_H$	32	FFFF $848C_H$	0	32
DMASS	DMAC Reload Source Address Register	DRSA2	$0000\ 0000_H$	32	FFFF 8490_H	0	32
DMASS	DMAC Reload Destination Address Register	DRDA2	$0000\ 0000_H$	32	FFFF 8494_H	0	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
DMASS	DMAC Reload Transfer Count Register	DRTC2	0000 0000$_H$	32	FFFF 8498$_H$	0	32
DMASS	DMAC Transfer Count Compare Register	DTCC2	0000 0000$_H$	32	FFFF 849C$_H$	0	32
DMASS	DMAC Channel Operation Enable Setting Register	DCEN2	0000 0000$_H$	32	FFFF 84A0$_H$	0	32
DMASS	DMAC Transfer Status Register	DCST2	0000 0000$_H$	32	FFFF 84A4$_H$	0	32
DMASS	DMAC Transfer Status Set Register	DCSTS2	0000 0000$_H$	32	FFFF 84A8$_H$	0	32
DMASS	DMAC Transfer Status Clear Register	DCSTC2	0000 0000$_H$	32	FFFF 84AC$_H$	0	32
DMASS	DTFR Setting Register	DTFR2	0000 0000$_H$	32	FFFF 84B0$_H$	0	32
DMASS	DTFR Transfer Request Status Register	DTFRRQ2	0000 0000$_H$	32	FFFF 84B4$_H$	0	32
DMASS	DTFR Transfer Request Clear Register	DTFRRQC2	0000 0000$_H$	32	FFFF 84B8$_H$	0	32
DMASS	DMAC Source Address Register	DSA3	0000 0000$_H$	32	FFFF 84C0$_H$	0	32
DMASS	DMAC Destination Address Register	DDA3	0000 0000$_H$	32	FFFF 84C4$_H$	0	32
DMASS	DMAC Transfer Count Register	DTC3	0000 0000$_H$	32	FFFF 84C8$_H$	0	32
DMASS	DMAC Transfer Control Register	DTCT3	0000 0000$_H$	32	FFFF 84CC$_H$	0	32
DMASS	DMAC Reload Source Address Register	DRSA3	0000 0000$_H$	32	FFFF 84D0$_H$	0	32
DMASS	DMAC Reload Destination Address Register	DRDA3	0000 0000$_H$	32	FFFF 84D4$_H$	0	32
DMASS	DMAC Reload Transfer Count Register	DRTC3	0000 0000$_H$	32	FFFF 84D8$_H$	0	32
DMASS	DMAC Transfer Count Compare Register	DTCC3	0000 0000$_H$	32	FFFF 84DC$_H$	0	32
DMASS	DMAC Channel Operation Enable Setting Register	DCEN3	0000 0000$_H$	32	FFFF 84E0$_H$	0	32
DMASS	DMAC Transfer Status Register	DCST3	0000 0000$_H$	32	FFFF 84E4$_H$	0	32
DMASS	DMAC Transfer Status Set Register	DCSTS3	0000 0000$_H$	32	FFFF 84E8$_H$	0	32
DMASS	DMAC Transfer Status Clear Register	DCSTC3	0000 0000$_H$	32	FFFF 84EC$_H$	0	32
DMASS	DTFR Setting Register	DTFR3	0000 0000$_H$	32	FFFF 84F0$_H$	0	32
DMASS	DTFR Transfer Request Status Register	DTFRRQ3	0000 0000$_H$	32	FFFF 84F4$_H$	0	32
DMASS	DTFR Transfer Request Clear Register	DTFRRQC3	0000 0000$_H$	32	FFFF 84F8$_H$	0	32
DMASS	DMAC Source Address Register	DSA4	0000 0000$_H$	32	FFFF 8500$_H$	0	32
DMASS	DMAC Destination Address Register	DDA4	0000 0000$_H$	32	FFFF 8504$_H$	0	32
DMASS	DMAC Transfer Count Register	DTC4	0000 0000$_H$	32	FFFF 8508$_H$	0	32
DMASS	DMAC Transfer Control Register	DTCT4	0000 0000$_H$	32	FFFF 850C$_H$	0	32
DMASS	DMAC Reload Source Address Register	DRSA4	0000 0000$_H$	32	FFFF 8510$_H$	0	32
DMASS	DMAC Reload Destination Address Register	DRDA4	0000 0000$_H$	32	FFFF 8514$_H$	0	32
DMASS	DMAC Reload Transfer Count Register	DRTC4	0000 0000$_H$	32	FFFF 8518$_H$	0	32
DMASS	DMAC Transfer Count Compare Register	DTCC4	0000 0000$_H$	32	FFFF 851C$_H$	0	32
DMASS	DMAC Channel Operation Enable Setting Register	DCEN4	0000 0000$_H$	32	FFFF 8520$_H$	0	32
DMASS	DMAC Transfer Status Register	DCST4	0000 0000$_H$	32	FFFF 8524$_H$	0	32
DMASS	DMAC Transfer Status Set Register	DCSTS4	0000 0000$_H$	32	FFFF 8528$_H$	0	32
DMASS	DMAC Transfer Status Clear Register	DCSTC4	0000 0000$_H$	32	FFFF 852C$_H$	0	32
DMASS	DTFR Setting Register	DTFR4	0000 0000$_H$	32	FFFF 8530$_H$	0	32
DMASS	DTFR Transfer Request Status Register	DTFRRQ4	0000 0000$_H$	32	FFFF 8534$_H$	0	32
DMASS	DTFR Transfer Request Clear Register	DTFRRQC4	0000 0000$_H$	32	FFFF 8538$_H$	0	32
DMASS	DMAC Source Address Register	DSA5	0000 0000$_H$	32	FFFF 8540$_H$	0	32
DMASS	DMAC Destination Address Register	DDA5	0000 0000$_H$	32	FFFF 8544$_H$	0	32
DMASS	DMAC Transfer Count Register	DTC5	0000 0000$_H$	32	FFFF 8548$_H$	0	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
DMASS	DMAC Transfer Control Register	DTCT5	0000 0000$_H$	32	FFFF 854C$_H$	0	32
DMASS	DMAC Reload Source Address Register	DRSA5	0000 0000$_H$	32	FFFF 8550$_H$	0	32
DMASS	DMAC Reload Destination Address Register	DRDA5	0000 0000$_H$	32	FFFF 8554$_H$	0	32
DMASS	DMAC Reload Transfer Count Register	DRTC5	0000 0000$_H$	32	FFFF 8558$_H$	0	32
DMASS	DMAC Transfer Count Compare Register	DTCC5	0000 0000$_H$	32	FFFF 855C$_H$	0	32
DMASS	DMAC Channel Operation Enable Setting Register	DCEN5	0000 0000$_H$	32	FFFF 8560$_H$	0	32
DMASS	DMAC Transfer Status Register	DCST5	0000 0000$_H$	32	FFFF 8564$_H$	0	32
DMASS	DMAC Transfer Status Set Register	DCSTS5	0000 0000$_H$	32	FFFF 8568$_H$	0	32
DMASS	DMAC Transfer Status Clear Register	DCSTC5	0000 0000$_H$	32	FFFF 856C$_H$	0	32
DMASS	DTFR Setting Register	DTFR5	0000 0000$_H$	32	FFFF 8570$_H$	0	32
DMASS	DTFR Transfer Request Status Register	DTFRRQ5	0000 0000$_H$	32	FFFF 8574$_H$	0	32
DMASS	DTFR Transfer Request Clear Register	DTFRRQC5	0000 0000$_H$	32	FFFF 8578$_H$	0	32
DMASS	DMAC Source Address Register	DSA6	0000 0000$_H$	32	FFFF 8580$_H$	0	32
DMASS	DMAC Destination Address Register	DDA6	0000 0000$_H$	32	FFFF 8584$_H$	0	32
DMASS	DMAC Transfer Count Register	DTC6	0000 0000$_H$	32	FFFF 8588$_H$	0	32
DMASS	DMAC Transfer Control Register	DTCT6	0000 0000$_H$	32	FFFF 858C$_H$	0	32
DMASS	DMAC Reload Source Address Register	DRSA6	0000 0000$_H$	32	FFFF 8590$_H$	0	32
DMASS	DMAC Reload Destination Address Register	DRDA6	0000 0000$_H$	32	FFFF 8594$_H$	0	32
DMASS	DMAC Reload Transfer Count Register	DRTC6	0000 0000$_H$	32	FFFF 8598$_H$	0	32
DMASS	DMAC Transfer Count Compare Register	DTCC6	0000 0000$_H$	32	FFFF 859C$_H$	0	32
DMASS	DMAC Channel Operation Enable Setting Register	DCEN6	0000 0000$_H$	32	FFFF 85A0$_H$	0	32
DMASS	DMAC Transfer Status Register	DCST6	0000 0000$_H$	32	FFFF 85A4$_H$	0	32
DMASS	DMAC Transfer Status Set Register	DCSTS6	0000 0000$_H$	32	FFFF 85A8$_H$	0	32
DMASS	DMAC Transfer Status Clear Register	DCSTC6	0000 0000$_H$	32	FFFF 85AC$_H$	0	32
DMASS	DTFR Setting Register	DTFR6	0000 0000$_H$	32	FFFF 85B0$_H$	0	32
DMASS	DTFR Transfer Request Status Register	DTFRRQ6	0000 0000$_H$	32	FFFF 85B4$_H$	0	32
DMASS	DTFR Transfer Request Clear Register	DTFRRQC6	0000 0000$_H$	32	FFFF 85B8$_H$	0	32
DMASS	DMAC Source Address Register	DSA7	0000 0000$_H$	32	FFFF 85C0$_H$	0	32
DMASS	DMAC Destination Address Register	DDA7	0000 0000$_H$	32	FFFF 85C4$_H$	0	32
DMASS	DMAC Transfer Count Register	DTC7	0000 0000$_H$	32	FFFF 85C8$_H$	0	32
DMASS	DMAC Transfer Control Register	DTCT7	0000 0000$_H$	32	FFFF 85CC$_H$	0	32
DMASS	DMAC Reload Source Address Register	DRSA7	0000 0000$_H$	32	FFFF 85D0$_H$	0	32
DMASS	DMAC Reload Destination Address Register	DRDA7	0000 0000$_H$	32	FFFF 85D4$_H$	0	32
DMASS	DMAC Reload Transfer Count Register	DRTC7	0000 0000$_H$	32	FFFF 85D8$_H$	0	32
DMASS	DMAC Transfer Count Compare Register	DTCC7	0000 0000$_H$	32	FFFF 85DC$_H$	0	32
DMASS	DMAC Channel Operation Enable Setting Register	DCEN7	0000 0000$_H$	32	FFFF 85E0$_H$	0	32
DMASS	DMAC Transfer Status Register	DCST7	0000 0000$_H$	32	FFFF 85E4$_H$	0	32
DMASS	DMAC Transfer Status Set Register	DCSTS7	0000 0000$_H$	32	FFFF 85E8$_H$	0	32
DMASS	DMAC Transfer Status Clear Register	DCSTC7	0000 0000$_H$	32	FFFF 85EC$_H$	0	32
DMASS	DTFR Setting Register	DTFR7	0000 0000$_H$	32	FFFF 85F0$_H$	0	32
DMASS	DTFR Transfer Request Status Register	DTFRRQ7	0000 0000$_H$	32	FFFF 85F4$_H$	0	32
DMASS	DTFR Transfer Request Clear Register	DTFRRQC7	0000 0000$_H$	32	FFFF 85F8$_H$	0	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
DMASS	DMAC Source Address Register	DSA8	0000 0000$_H$	32	FFFF 8600$_H$	0	32
DMASS	DMAC Destination Address Register	DDA8	0000 0000$_H$	32	FFFF 8604$_H$	0	32
DMASS	DMAC Transfer Count Register	DTC8	0000 0000$_H$	32	FFFF 8608$_H$	0	32
DMASS	DMAC Transfer Control Register	DTCT8	0000 0000$_H$	32	FFFF 860C$_H$	0	32
DMASS	DMAC Reload Source Address Register	DRSA8	0000 0000$_H$	32	FFFF 8610$_H$	0	32
DMASS	DMAC Reload Destination Address Register	DRDA8	0000 0000$_H$	32	FFFF 8614$_H$	0	32
DMASS	DMAC Reload Transfer Count Register	DRTC8	0000 0000$_H$	32	FFFF 8618$_H$	0	32
DMASS	DMAC Transfer Count Compare Register	DTCC8	0000 0000$_H$	32	FFFF 861C$_H$	0	32
DMASS	DMAC Channel Operation Enable Setting Register	DCEN8	0000 0000$_H$	32	FFFF 8620$_H$	0	32
DMASS	DMAC Transfer Status Register	DCST8	0000 0000$_H$	32	FFFF 8624$_H$	0	32
DMASS	DMAC Transfer Status Set Register	DCSTS8	0000 0000$_H$	32	FFFF 8628$_H$	0	32
DMASS	DMAC Transfer Status Clear Register	DCSTC8	0000 0000$_H$	32	FFFF 862C$_H$	0	32
DMASS	DTFR Setting Register	DTFR8	0000 0000$_H$	32	FFFF 8630$_H$	0	32
DMASS	DTFR Transfer Request Status Register	DTFRRQ8	0000 0000$_H$	32	FFFF 8634$_H$	0	32
DMASS	DTFR Transfer Request Clear Register	DTFRRQC8	0000 0000$_H$	32	FFFF 8638$_H$	0	32
DMASS	DMAC Source Address Register	DSA9	0000 0000$_H$	32	FFFF 8640$_H$	0	32
DMASS	DMAC Destination Address Register	DDA9	0000 0000$_H$	32	FFFF 8644$_H$	0	32
DMASS	DMAC Transfer Count Register	DTC9	0000 0000$_H$	32	FFFF 8648$_H$	0	32
DMASS	DMAC Transfer Control Register	DTCT9	0000 0000$_H$	32	FFFF 864C$_H$	0	32
DMASS	DMAC Reload Source Address Register	DRSA9	0000 0000$_H$	32	FFFF 8650$_H$	0	32
DMASS	DMAC Reload Destination Address Register	DRDA9	0000 0000$_H$	32	FFFF 8654$_H$	0	32
DMASS	DMAC Reload Transfer Count Register	DRTC9	0000 0000$_H$	32	FFFF 8658$_H$	0	32
DMASS	DMAC Transfer Count Compare Register	DTCC9	0000 0000$_H$	32	FFFF 865C$_H$	0	32
DMASS	DMAC Channel Operation Enable Setting Register	DCEN9	0000 0000$_H$	32	FFFF 8660$_H$	0	32
DMASS	DMAC Transfer Status Register	DCST9	0000 0000$_H$	32	FFFF 8664$_H$	0	32
DMASS	DMAC Transfer Status Set Register	DCSTS9	0000 0000$_H$	32	FFFF 8668$_H$	0	32
DMASS	DMAC Transfer Status Clear Register	DCSTC9	0000 0000$_H$	32	FFFF 866C$_H$	0	32
DMASS	DTFR Setting Register	DTFR9	0000 0000$_H$	32	FFFF 8670$_H$	0	32
DMASS	DTFR Transfer Request Status Register	DTFRRQ9	0000 0000$_H$	32	FFFF 8674$_H$	0	32
DMASS	DTFR Transfer Request Clear Register	DTFRRQC9	0000 0000$_H$	32	FFFF 8678$_H$	0	32
DMASS	DMAC Source Address Register	DSA10	0000 0000$_H$	32	FFFF 8680$_H$	0	32
DMASS	DMAC Destination Address Register	DDA10	0000 0000$_H$	32	FFFF 8684$_H$	0	32
DMASS	DMAC Transfer Count Register	DTC10	0000 0000$_H$	32	FFFF 8688$_H$	0	32
DMASS	DMAC Transfer Control Register	DTCT10	0000 0000$_H$	32	FFFF 868C$_H$	0	32
DMASS	DMAC Reload Source Address Register	DRSA10	0000 0000$_H$	32	FFFF 8690$_H$	0	32
DMASS	DMAC Reload Destination Address Register	DRDA10	0000 0000$_H$	32	FFFF 8694$_H$	0	32
DMASS	DMAC Reload Transfer Count Register	DRTC10	0000 0000$_H$	32	FFFF 8698$_H$	0	32
DMASS	DMAC Transfer Count Compare Register	DTCC10	0000 0000$_H$	32	FFFF 869C$_H$	0	32
DMASS	DMAC Channel Operation Enable Setting Register	DCEN10	0000 0000$_H$	32	FFFF 86A0$_H$	0	32
DMASS	DMAC Transfer Status Register	DCST10	0000 0000$_H$	32	FFFF 86A4$_H$	0	32
DMASS	DMAC Transfer Status Set Register	DCSTS10	0000 0000$_H$	32	FFFF 86A8$_H$	0	32
DMASS	DMAC Transfer Status Clear Register	DCSTC10	0000 0000$_H$	32	FFFF 86AC$_H$	0	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
DMASS	DTFR Setting Register	DTFR10	0000 0000$_H$	32	FFFF 86B0$_H$	0	32
DMASS	DTFR Transfer Request Status Register	DTFRRQ10	0000 0000$_H$	32	FFFF 86B4$_H$	0	32
DMASS	DTFR Transfer Request Clear Register	DTFRRQC10	0000 0000$_H$	32	FFFF 86B8$_H$	0	32
DMASS	DMAC Source Address Register	DSA11	0000 0000$_H$	32	FFFF 86C0$_H$	0	32
DMASS	DMAC Destination Address Register	DDA11	0000 0000$_H$	32	FFFF 86C4$_H$	0	32
DMASS	DMAC Transfer Count Register	DTC11	0000 0000$_H$	32	FFFF 86C8$_H$	0	32
DMASS	DMAC Transfer Control Register	DTCT11	0000 0000$_H$	32	FFFF 86CC$_H$	0	32
DMASS	DMAC Reload Source Address Register	DRSA11	0000 0000$_H$	32	FFFF 86D0$_H$	0	32
DMASS	DMAC Reload Destination Address Register	DRDA11	0000 0000$_H$	32	FFFF 86D4$_H$	0	32
DMASS	DMAC Reload Transfer Count Register	DRTC11	0000 0000$_H$	32	FFFF 86D8$_H$	0	32
DMASS	DMAC Transfer Count Compare Register	DTCC11	0000 0000$_H$	32	FFFF 86DC$_H$	0	32
DMASS	DMAC Channel Operation Enable Setting Register	DCEN11	0000 0000$_H$	32	FFFF 86E0$_H$	0	32
DMASS	DMAC Transfer Status Register	DCST11	0000 0000$_H$	32	FFFF 86E4$_H$	0	32
DMASS	DMAC Transfer Status Set Register	DCSTS11	0000 0000$_H$	32	FFFF 86E8$_H$	0	32
DMASS	DMAC Transfer Status Clear Register	DCSTC11	0000 0000$_H$	32	FFFF 86EC$_H$	0	32
DMASS	DTFR Setting Register	DTFR11	0000 0000$_H$	32	FFFF 86F0$_H$	0	32
DMASS	DTFR Transfer Request Status Register	DTFRRQ11	0000 0000$_H$	32	FFFF 86F4$_H$	0	32
DMASS	DTFR Transfer Request Clear Register	DTFRRQC11	0000 0000$_H$	32	FFFF 86F8$_H$	0	32
DMASS	DMAC Source Address Register	DSA12	0000 0000$_H$	32	FFFF 8700$_H$	0	32
DMASS	DMAC Destination Address Register	DDA12	0000 0000$_H$	32	FFFF 8704$_H$	0	32
DMASS	DMAC Transfer Count Register	DTC12	0000 0000$_H$	32	FFFF 8708$_H$	0	32
DMASS	DMAC Transfer Control Register	DTCT12	0000 0000$_H$	32	FFFF 870C$_H$	0	32
DMASS	DMAC Reload Source Address Register	DRSA12	0000 0000$_H$	32	FFFF 8710$_H$	0	32
DMASS	DMAC Reload Destination Address Register	DRDA12	0000 0000$_H$	32	FFFF 8714$_H$	0	32
DMASS	DMAC Reload Transfer Count Register	DRTC12	0000 0000$_H$	32	FFFF 8718$_H$	0	32
DMASS	DMAC Transfer Count Compare Register	DTCC12	0000 0000$_H$	32	FFFF 871C$_H$	0	32
DMASS	DMAC Channel Operation Enable Setting Register	DCEN12	0000 0000$_H$	32	FFFF 8720$_H$	0	32
DMASS	DMAC Transfer Status Register	DCST12	0000 0000$_H$	32	FFFF 8724$_H$	0	32
DMASS	DMAC Transfer Status Set Register	DCSTS12	0000 0000$_H$	32	FFFF 8728$_H$	0	32
DMASS	DMAC Transfer Status Clear Register	DCSTC12	0000 0000$_H$	32	FFFF 872C$_H$	0	32
DMASS	DTFR Setting Register	DTFR12	0000 0000$_H$	32	FFFF 8730$_H$	0	32
DMASS	DTFR Transfer Request Status Register	DTFRRQ12	0000 0000$_H$	32	FFFF 8734$_H$	0	32
DMASS	DTFR Transfer Request Clear Register	DTFRRQC12	0000 0000$_H$	32	FFFF 8738$_H$	0	32
DMASS	DMAC Source Address Register	DSA13	0000 0000$_H$	32	FFFF 8740$_H$	0	32
DMASS	DMAC Destination Address Register	DDA13	0000 0000$_H$	32	FFFF 8744$_H$	0	32
DMASS	DMAC Transfer Count Register	DTC13	0000 0000$_H$	32	FFFF 8748$_H$	0	32
DMASS	DMAC Transfer Control Register	DTCT13	0000 0000$_H$	32	FFFF 874C$_H$	0	32
DMASS	DMAC Reload Source Address Register	DRSA13	0000 0000$_H$	32	FFFF 8750$_H$	0	32
DMASS	DMAC Reload Destination Address Register	DRDA13	0000 0000$_H$	32	FFFF 8754$_H$	0	32
DMASS	DMAC Reload Transfer Count Register	DRTC13	0000 0000$_H$	32	FFFF 8758$_H$	0	32
DMASS	DMAC Transfer Count Compare Register	DTCC13	0000 0000$_H$	32	FFFF 875C$_H$	0	32
DMASS	DMAC Channel Operation Enable Setting Register	DCEN13	0000 0000$_H$	32	FFFF 8760$_H$	0	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
DMASS	DMAC Transfer Status Register	DCST13	0000 0000$_H$	32	FFFF 8764$_H$	0	32
DMASS	DMAC Transfer Status Set Register	DCSTS13	0000 0000$_H$	32	FFFF 8768$_H$	0	32
DMASS	DMAC Transfer Status Clear Register	DCSTC13	0000 0000$_H$	32	FFFF 876C$_H$	0	32
DMASS	DTFR Setting Register	DTFR13	0000 0000$_H$	32	FFFF 8770$_H$	0	32
DMASS	DTFR Transfer Request Status Register	DTFRRQ13	0000 0000$_H$	32	FFFF 8774$_H$	0	32
DMASS	DTFR Transfer Request Clear Register	DTFRRQC13	0000 0000$_H$	32	FFFF 8778$_H$	0	32
DMASS	DMAC Source Address Register	DSA14	0000 0000$_H$	32	FFFF 8780$_H$	0	32
DMASS	DMAC Destination Address Register	DDA14	0000 0000$_H$	32	FFFF 8784$_H$	0	32
DMASS	DMAC Transfer Count Register	DTC14	0000 0000$_H$	32	FFFF 8788$_H$	0	32
DMASS	DMAC Transfer Control Register	DTCT14	0000 0000$_H$	32	FFFF 878C$_H$	0	32
DMASS	DMAC Reload Source Address Register	DRSA14	0000 0000$_H$	32	FFFF 8790$_H$	0	32
DMASS	DMAC Reload Destination Address Register	DRDA14	0000 0000$_H$	32	FFFF 8794$_H$	0	32
DMASS	DMAC Reload Transfer Count Register	DRTC14	0000 0000$_H$	32	FFFF 8798$_H$	0	32
DMASS	DMAC Transfer Count Compare Register	DTCC14	0000 0000$_H$	32	FFFF 879C$_H$	0	32
DMASS	DMAC Channel Operation Enable Setting Register	DCEN14	0000 0000$_H$	32	FFFF 87A0$_H$	0	32
DMASS	DMAC Transfer Status Register	DCST14	0000 0000$_H$	32	FFFF 87A4$_H$	0	32
DMASS	DMAC Transfer Status Set Register	DCSTS14	0000 0000$_H$	32	FFFF 87A8$_H$	0	32
DMASS	DMAC Transfer Status Clear Register	DCSTC14	0000 0000$_H$	32	FFFF 87AC$_H$	0	32
DMASS	DTFR Setting Register	DTFR14	0000 0000$_H$	32	FFFF 87B0$_H$	0	32
DMASS	DTFR Transfer Request Status Register	DTFRRQ14	0000 0000$_H$	32	FFFF 87B4$_H$	0	32
DMASS	DTFR Transfer Request Clear Register	DTFRRQC14	0000 0000$_H$	32	FFFF 87B8$_H$	0	32
DMASS	DMAC Source Address Register	DSA15	0000 0000$_H$	32	FFFF 87C0$_H$	0	32
DMASS	DMAC Destination Address Register	DDA15	0000 0000$_H$	32	FFFF 87C4$_H$	0	32
DMASS	DMAC Transfer Count Register	DTC15	0000 0000$_H$	32	FFFF 87C8$_H$	0	32
DMASS	DMAC Transfer Control Register	DTCT15	0000 0000$_H$	32	FFFF 87CC$_H$	0	32
DMASS	DMAC Reload Source Address Register	DRSA15	0000 0000$_H$	32	FFFF 87D0$_H$	0	32
DMASS	DMAC Reload Destination Address Register	DRDA15	0000 0000$_H$	32	FFFF 87D4$_H$	0	32
DMASS	DMAC Reload Transfer Count Register	DRTC15	0000 0000$_H$	32	FFFF 87D8$_H$	0	32
DMASS	DMAC Transfer Count Compare Register	DTCC15	0000 0000$_H$	32	FFFF 87DC$_H$	0	32
DMASS	DMAC Channel Operation Enable Setting Register	DCEN15	0000 0000$_H$	32	FFFF 87E0$_H$	0	32
DMASS	DMAC Transfer Status Register	DCST15	0000 0000$_H$	32	FFFF 87E4$_H$	0	32
DMASS	DMAC Transfer Status Set Register	DCSTS15	0000 0000$_H$	32	FFFF 87E8$_H$	0	32
DMASS	DMAC Transfer Status Clear Register	DCSTC15	0000 0000$_H$	32	FFFF 87EC$_H$	0	32
DMASS	DTFR Setting Register	DTFR15	0000 0000$_H$	32	FFFF 87F0$_H$	0	32
DMASS	DTFR Transfer Request Status Register	DTFRRQ15	0000 0000$_H$	32	FFFF 87F4$_H$	0	32
DMASS	DTFR Transfer Request Clear Register	DTFRRQC15	0000 0000$_H$	32	FFFF 87F8$_H$	0	32
DMASS	DTS Source Address Register	DTSA000	XXXX XXXX$_H$	32	FFFF 9000$_H$	0	32
DMASS	DTS Destination Address Register	DTDA000	XXXX XXXX$_H$	32	FFFF 9004$_H$	0	32
DMASS	DTS Transfer Count Register	DTTC000	XXXX XXXX$_H$	32	FFFF 9008$_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT000	XXXX XXXX$_H$	32	FFFF 900C$_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA000	XXXX XXXX$_H$	32	FFFF 9010$_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA000	XXXX XXXX$_H$	32	FFFF 9014$_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC000	XXXX XXXX$_H$	32	FFFF 9018$_H$	0	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
DMASS	DTS Transfer Count Compare Register	DTTCC000	XXXX $XXXX_H$	32	FFFF $901C_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL000	$0000\ 0000_H$	32	FFFF 9020_H	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST000	$0000\ 0000_H$	32	FFFF 9024_H	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS000	$0000\ 0000_H$	32	FFFF 9028_H	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC000	$0000\ 0000_H$	32	FFFF $902C_H$	0	32
DMASS	DTS Source Address Register	DTSA001	XXXX $XXXX_H$	32	FFFF 9040_H	0	32
DMASS	DTS Destination Address Register	DTDA001	XXXX $XXXX_H$	32	FFFF 9044_H	0	32
DMASS	DTS Transfer Count Register	DTTC001	XXXX $XXXX_H$	32	FFFF 9048_H	0	32
DMASS	DTS Transfer Control Register	DTTCT001	XXXX $XXXX_H$	32	FFFF $904C_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA001	XXXX $XXXX_H$	32	FFFF 9050_H	0	32
DMASS	DTS Reload Destination Address Register	DTRDA001	XXXX $XXXX_H$	32	FFFF 9054_H	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC001	XXXX $XXXX_H$	32	FFFF 9058_H	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC001	XXXX $XXXX_H$	32	FFFF $905C_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL001	$0000\ 0000_H$	32	FFFF 9060_H	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST001	$0000\ 0000_H$	32	FFFF 9064_H	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS001	$0000\ 0000_H$	32	FFFF 9068_H	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC001	$0000\ 0000_H$	32	FFFF $906C_H$	0	32
DMASS	DTS Source Address Register	DTSA002	XXXX $XXXX_H$	32	FFFF 9080_H	0	32
DMASS	DTS Destination Address Register	DTDA002	XXXX $XXXX_H$	32	FFFF 9084_H	0	32
DMASS	DTS Transfer Count Register	DTTC002	XXXX $XXXX_H$	32	FFFF 9088_H	0	32
DMASS	DTS Transfer Control Register	DTTCT002	XXXX $XXXX_H$	32	FFFF $908C_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA002	XXXX $XXXX_H$	32	FFFF 9090_H	0	32
DMASS	DTS Reload Destination Address Register	DTRDA002	XXXX $XXXX_H$	32	FFFF 9094_H	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC002	XXXX $XXXX_H$	32	FFFF 9098_H	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC002	XXXX $XXXX_H$	32	FFFF $909C_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL002	$0000\ 0000_H$	32	FFFF $90A0_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST002	$0000\ 0000_H$	32	FFFF $90A4_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS002	$0000\ 0000_H$	32	FFFF $90A8_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC002	$0000\ 0000_H$	32	FFFF $90AC_H$	0	32
DMASS	DTS Source Address Register	DTSA003	XXXX $XXXX_H$	32	FFFF $90C0_H$	0	32
DMASS	DTS Destination Address Register	DTDA003	XXXX $XXXX_H$	32	FFFF $90C4_H$	0	32
DMASS	DTS Transfer Count Register	DTTC003	XXXX $XXXX_H$	32	FFFF $90C8_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT003	XXXX $XXXX_H$	32	FFFF $90CC_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA003	XXXX $XXXX_H$	32	FFFF $90D0_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA003	XXXX $XXXX_H$	32	FFFF $90D4_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC003	XXXX $XXXX_H$	32	FFFF $90D8_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC003	XXXX $XXXX_H$	32	FFFF $90DC_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL003	$0000\ 0000_H$	32	FFFF $90E0_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST003	$0000\ 0000_H$	32	FFFF $90E4_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS003	$0000\ 0000_H$	32	FFFF $90E8_H$	0	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
DMASS	DTSFSL Transfer Request Clear Register	DTFSC003	0000 0000_H	32	FFFF $90EC_H$	0	32
DMASS	DTS Source Address Register	DTSA004	XXXX $XXXX_H$	32	FFFF 9100_H	0	32
DMASS	DTS Destination Address Register	DTDA004	XXXX $XXXX_H$	32	FFFF 9104_H	0	32
DMASS	DTS Transfer Count Register	DTTC004	XXXX $XXXX_H$	32	FFFF 9108_H	0	32
DMASS	DTS Transfer Control Register	DTTCT004	XXXX $XXXX_H$	32	FFFF $910C_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA004	XXXX $XXXX_H$	32	FFFF 9110_H	0	32
DMASS	DTS Reload Destination Address Register	DTRDA004	XXXX $XXXX_H$	32	FFFF 9114_H	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC004	XXXX $XXXX_H$	32	FFFF 9118_H	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC004	XXXX $XXXX_H$	32	FFFF $911C_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL004	0000 0000_H	32	FFFF 9120_H	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST004	0000 0000_H	32	FFFF 9124_H	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS004	0000 0000_H	32	FFFF 9128_H	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC004	0000 0000_H	32	FFFF $912C_H$	0	32
DMASS	DTS Source Address Register	DTSA005	XXXX $XXXX_H$	32	FFFF 9140_H	0	32
DMASS	DTS Destination Address Register	DTDA005	XXXX $XXXX_H$	32	FFFF 9144_H	0	32
DMASS	DTS Transfer Count Register	DTTC005	XXXX $XXXX_H$	32	FFFF 9148_H	0	32
DMASS	DTS Transfer Control Register	DTTCT005	XXXX $XXXX_H$	32	FFFF $914C_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA005	XXXX $XXXX_H$	32	FFFF 9150_H	0	32
DMASS	DTS Reload Destination Address Register	DTRDA005	XXXX $XXXX_H$	32	FFFF 9154_H	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC005	XXXX $XXXX_H$	32	FFFF 9158_H	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC005	XXXX $XXXX_H$	32	FFFF $915C_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL005	0000 0000_H	32	FFFF 9160_H	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST005	0000 0000_H	32	FFFF 9164_H	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS005	0000 0000_H	32	FFFF 9168_H	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC005	0000 0000_H	32	FFFF $916C_H$	0	32
DMASS	DTS Source Address Register	DTSA006	XXXX $XXXX_H$	32	FFFF 9180_H	0	32
DMASS	DTS Destination Address Register	DTDA006	XXXX $XXXX_H$	32	FFFF 9184_H	0	32
DMASS	DTS Transfer Count Register	DTTC006	XXXX $XXXX_H$	32	FFFF 9188_H	0	32
DMASS	DTS Transfer Control Register	DTTCT006	XXXX $XXXX_H$	32	FFFF $918C_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA006	XXXX $XXXX_H$	32	FFFF 9190_H	0	32
DMASS	DTS Reload Destination Address Register	DTRDA006	XXXX $XXXX_H$	32	FFFF 9194_H	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC006	XXXX $XXXX_H$	32	FFFF 9198_H	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC006	XXXX $XXXX_H$	32	FFFF $919C_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL006	0000 0000_H	32	FFFF $91A0_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST006	0000 0000_H	32	FFFF $91A4_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS006	0000 0000_H	32	FFFF $91A8_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC006	0000 0000_H	32	FFFF $91AC_H$	0	32
DMASS	DTS Source Address Register	DTSA007	XXXX $XXXX_H$	32	FFFF $91C0_H$	0	32
DMASS	DTS Destination Address Register	DTDA007	XXXX $XXXX_H$	32	FFFF $91C4_H$	0	32
DMASS	DTS Transfer Count Register	DTTC007	XXXX $XXXX_H$	32	FFFF $91C8_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT007	XXXX $XXXX_H$	32	FFFF $91CC_H$	0	32

(147/201)

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
DMASS	DTS Reload Source Address Register	DTRSA007	XXXX XXXX$_H$	32	FFFF 91D0$_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA007	XXXX XXXX$_H$	32	FFFF 91D4$_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC007	XXXX XXXX$_H$	32	FFFF 91D8$_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC007	XXXX XXXX$_H$	32	FFFF 91DC$_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL007	0000 0000$_H$	32	FFFF 91E0$_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST007	0000 0000$_H$	32	FFFF 91E4$_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS007	0000 0000$_H$	32	FFFF 91E8$_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC007	0000 0000$_H$	32	FFFF 91EC$_H$	0	32
DMASS	DTS Source Address Register	DTSA008	XXXX XXXX$_H$	32	FFFF 9200$_H$	0	32
DMASS	DTS Destination Address Register	DTDA008	XXXX XXXX$_H$	32	FFFF 9204$_H$	0	32
DMASS	DTS Transfer Count Register	DTTC008	XXXX XXXX$_H$	32	FFFF 9208$_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT008	XXXX XXXX$_H$	32	FFFF 920C$_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA008	XXXX XXXX$_H$	32	FFFF 9210$_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA008	XXXX XXXX$_H$	32	FFFF 9214$_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC008	XXXX XXXX$_H$	32	FFFF 9218$_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC008	XXXX XXXX$_H$	32	FFFF 921C$_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL008	0000 0000$_H$	32	FFFF 9220$_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST008	0000 0000$_H$	32	FFFF 9224$_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS008	0000 0000$_H$	32	FFFF 9228$_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC008	0000 0000$_H$	32	FFFF 922C$_H$	0	32
DMASS	DTS Source Address Register	DTSA009	XXXX XXXX$_H$	32	FFFF 9240$_H$	0	32
DMASS	DTS Destination Address Register	DTDA009	XXXX XXXX$_H$	32	FFFF 9244$_H$	0	32
DMASS	DTS Transfer Count Register	DTTC009	XXXX XXXX$_H$	32	FFFF 9248$_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT009	XXXX XXXX$_H$	32	FFFF 924C$_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA009	XXXX XXXX$_H$	32	FFFF 9250$_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA009	XXXX XXXX$_H$	32	FFFF 9254$_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC009	XXXX XXXX$_H$	32	FFFF 9258$_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC009	XXXX XXXX$_H$	32	FFFF 925C$_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL009	0000 0000$_H$	32	FFFF 9260$_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST009	0000 0000$_H$	32	FFFF 9264$_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS009	0000 0000$_H$	32	FFFF 9268$_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC009	0000 0000$_H$	32	FFFF 926C$_H$	0	32
DMASS	DTS Source Address Register	DTSA010	XXXX XXXX$_H$	32	FFFF 9280$_H$	0	32
DMASS	DTS Destination Address Register	DTDA010	XXXX XXXX$_H$	32	FFFF 9284$_H$	0	32
DMASS	DTS Transfer Count Register	DTTC010	XXXX XXXX$_H$	32	FFFF 9288$_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT010	XXXX XXXX$_H$	32	FFFF 928C$_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA010	XXXX XXXX$_H$	32	FFFF 9290$_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA010	XXXX XXXX$_H$	32	FFFF 9294$_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC010	XXXX XXXX$_H$	32	FFFF 9298$_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC010	XXXX XXXX$_H$	32	FFFF 929C$_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL010	0000 0000$_H$	32	FFFF 92A0$_H$	0	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
DMASS	DTSFSL Transfer Request Status Register	DTFST010	0000 0000_H	32	FFFF $92A4_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS010	0000 0000_H	32	FFFF $92A8_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC010	0000 0000_H	32	FFFF $92AC_H$	0	32
DMASS	DTS Source Address Register	DTSA011	XXXX $XXXX_H$	32	FFFF $92C0_H$	0	32
DMASS	DTS Destination Address Register	DTDA011	XXXX $XXXX_H$	32	FFFF $92C4_H$	0	32
DMASS	DTS Transfer Count Register	DTTC011	XXXX $XXXX_H$	32	FFFF $92C8_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT011	XXXX $XXXX_H$	32	FFFF $92CC_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA011	XXXX $XXXX_H$	32	FFFF $92D0_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA011	XXXX $XXXX_H$	32	FFFF $92D4_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC011	XXXX $XXXX_H$	32	FFFF $92D8_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC011	XXXX $XXXX_H$	32	FFFF $92DC_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL011	0000 0000_H	32	FFFF $92E0_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST011	0000 0000_H	32	FFFF $92E4_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS011	0000 0000_H	32	FFFF $92E8_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC011	0000 0000_H	32	FFFF $92EC_H$	0	32
DMASS	DTS Source Address Register	DTSA012	XXXX $XXXX_H$	32	FFFF 9300_H	0	32
DMASS	DTS Destination Address Register	DTDA012	XXXX $XXXX_H$	32	FFFF 9304_H	0	32
DMASS	DTS Transfer Count Register	DTTC012	XXXX $XXXX_H$	32	FFFF 9308_H	0	32
DMASS	DTS Transfer Control Register	DTTCT012	XXXX $XXXX_H$	32	FFFF $930C_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA012	XXXX $XXXX_H$	32	FFFF 9310_H	0	32
DMASS	DTS Reload Destination Address Register	DTRDA012	XXXX $XXXX_H$	32	FFFF 9314_H	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC012	XXXX $XXXX_H$	32	FFFF 9318_H	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC012	XXXX $XXXX_H$	32	FFFF $931C_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL012	0000 0000_H	32	FFFF 9320_H	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST012	0000 0000_H	32	FFFF 9324_H	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS012	0000 0000_H	32	FFFF 9328_H	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC012	0000 0000_H	32	FFFF $932C_H$	0	32
DMASS	DTS Source Address Register	DTSA013	XXXX $XXXX_H$	32	FFFF 9340_H	0	32
DMASS	DTS Destination Address Register	DTDA013	XXXX $XXXX_H$	32	FFFF 9344_H	0	32
DMASS	DTS Transfer Count Register	DTTC013	XXXX $XXXX_H$	32	FFFF 9348_H	0	32
DMASS	DTS Transfer Control Register	DTTCT013	XXXX $XXXX_H$	32	FFFF $934C_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA013	XXXX $XXXX_H$	32	FFFF 9350_H	0	32
DMASS	DTS Reload Destination Address Register	DTRDA013	XXXX $XXXX_H$	32	FFFF 9354_H	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC013	XXXX $XXXX_H$	32	FFFF 9358_H	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC013	XXXX $XXXX_H$	32	FFFF $935C_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL013	0000 0000_H	32	FFFF 9360_H	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST013	0000 0000_H	32	FFFF 9364_H	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS013	0000 0000_H	32	FFFF 9368_H	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC013	0000 0000_H	32	FFFF $936C_H$	0	32
DMASS	DTS Source Address Register	DTSA014	XXXX $XXXX_H$	32	FFFF 9380_H	0	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
DMASS	DTS Destination Address Register	DTDA014	XXXX XXXX$_H$	32	FFFF 9384$_H$	0	32
DMASS	DTS Transfer Count Register	DTTC014	XXXX XXXX$_H$	32	FFFF 9388$_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT014	XXXX XXXX$_H$	32	FFFF 938C$_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA014	XXXX XXXX$_H$	32	FFFF 9390$_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA014	XXXX XXXX$_H$	32	FFFF 9394$_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC014	XXXX XXXX$_H$	32	FFFF 9398$_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC014	XXXX XXXX$_H$	32	FFFF 939C$_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL014	0000 0000$_H$	32	FFFF 93A0$_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST014	0000 0000$_H$	32	FFFF 93A4$_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS014	0000 0000$_H$	32	FFFF 93A8$_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC014	0000 0000$_H$	32	FFFF 93AC$_H$	0	32
DMASS	DTS Source Address Register	DTSA015	XXXX XXXX$_H$	32	FFFF 93C0$_H$	0	32
DMASS	DTS Destination Address Register	DTDA015	XXXX XXXX$_H$	32	FFFF 93C4$_H$	0	32
DMASS	DTS Transfer Count Register	DTTC015	XXXX XXXX$_H$	32	FFFF 93C8$_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT015	XXXX XXXX$_H$	32	FFFF 93CC$_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA015	XXXX XXXX$_H$	32	FFFF 93D0$_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA015	XXXX XXXX$_H$	32	FFFF 93D4$_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC015	XXXX XXXX$_H$	32	FFFF 93D8$_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC015	XXXX XXXX$_H$	32	FFFF 93DC$_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL015	0000 0000$_H$	32	FFFF 93E0$_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST015	0000 0000$_H$	32	FFFF 93E4$_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS015	0000 0000$_H$	32	FFFF 93E8$_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC015	0000 0000$_H$	32	FFFF 93EC$_H$	0	32
DMASS	DTS Source Address Register	DTSA016	XXXX XXXX$_H$	32	FFFF 9400$_H$	0	32
DMASS	DTS Destination Address Register	DTDA016	XXXX XXXX$_H$	32	FFFF 9404$_H$	0	32
DMASS	DTS Transfer Count Register	DTTC016	XXXX XXXX$_H$	32	FFFF 9408$_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT016	XXXX XXXX$_H$	32	FFFF 940C$_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA016	XXXX XXXX$_H$	32	FFFF 9410$_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA016	XXXX XXXX$_H$	32	FFFF 9414$_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC016	XXXX XXXX$_H$	32	FFFF 9418$_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC016	XXXX XXXX$_H$	32	FFFF 941C$_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL016	0000 0000$_H$	32	FFFF 9420$_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST016	0000 0000$_H$	32	FFFF 9424$_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS016	0000 0000$_H$	32	FFFF 9428$_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC016	0000 0000$_H$	32	FFFF 942C$_H$	0	32
DMASS	DTS Source Address Register	DTSA017	XXXX XXXX$_H$	32	FFFF 9440$_H$	0	32
DMASS	DTS Destination Address Register	DTDA017	XXXX XXXX$_H$	32	FFFF 9444$_H$	0	32
DMASS	DTS Transfer Count Register	DTTC017	XXXX XXXX$_H$	32	FFFF 9448$_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT017	XXXX XXXX$_H$	32	FFFF 944C$_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA017	XXXX XXXX$_H$	32	FFFF 9450$_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA017	XXXX XXXX$_H$	32	FFFF 9454$_H$	0	32

(150/201)

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
DMASS	DTS Reload Transfer Count Register	DTRTC017	XXXX XXXX$_H$	32	FFFF 9458$_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC017	XXXX XXXX$_H$	32	FFFF 945C$_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL017	0000 0000$_H$	32	FFFF 9460$_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST017	0000 0000$_H$	32	FFFF 9464$_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS017	0000 0000$_H$	32	FFFF 9468$_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC017	0000 0000$_H$	32	FFFF 946C$_H$	0	32
DMASS	DTS Source Address Register	DTSA018	XXXX XXXX$_H$	32	FFFF 9480$_H$	0	32
DMASS	DTS Destination Address Register	DTDA018	XXXX XXXX$_H$	32	FFFF 9484$_H$	0	32
DMASS	DTS Transfer Count Register	DTTC018	XXXX XXXX$_H$	32	FFFF 9488$_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT018	XXXX XXXX$_H$	32	FFFF 948C$_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA018	XXXX XXXX$_H$	32	FFFF 9490$_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA018	XXXX XXXX$_H$	32	FFFF 9494$_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC018	XXXX XXXX$_H$	32	FFFF 9498$_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC018	XXXX XXXX$_H$	32	FFFF 949C$_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL018	0000 0000$_H$	32	FFFF 94A0$_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST018	0000 0000$_H$	32	FFFF 94A4$_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS018	0000 0000$_H$	32	FFFF 94A8$_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC018	0000 0000$_H$	32	FFFF 94AC$_H$	0	32
DMASS	DTS Source Address Register	DTSA019	XXXX XXXX$_H$	32	FFFF 94C0$_H$	0	32
DMASS	DTS Destination Address Register	DTDA019	XXXX XXXX$_H$	32	FFFF 94C4$_H$	0	32
DMASS	DTS Transfer Count Register	DTTC019	XXXX XXXX$_H$	32	FFFF 94C8$_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT019	XXXX XXXX$_H$	32	FFFF 94CC$_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA019	XXXX XXXX$_H$	32	FFFF 94D0$_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA019	XXXX XXXX$_H$	32	FFFF 94D4$_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC019	XXXX XXXX$_H$	32	FFFF 94D8$_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC019	XXXX XXXX$_H$	32	FFFF 94DC$_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL019	0000 0000$_H$	32	FFFF 94E0$_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST019	0000 0000$_H$	32	FFFF 94E4$_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS019	0000 0000$_H$	32	FFFF 94E8$_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC019	0000 0000$_H$	32	FFFF 94EC$_H$	0	32
DMASS	DTS Source Address Register	DTSA020	XXXX XXXX$_H$	32	FFFF 9500$_H$	0	32
DMASS	DTS Destination Address Register	DTDA020	XXXX XXXX$_H$	32	FFFF 9504$_H$	0	32
DMASS	DTS Transfer Count Register	DTTC020	XXXX XXXX$_H$	32	FFFF 9508$_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT020	XXXX XXXX$_H$	32	FFFF 950C$_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA020	XXXX XXXX$_H$	32	FFFF 9510$_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA020	XXXX XXXX$_H$	32	FFFF 9514$_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC020	XXXX XXXX$_H$	32	FFFF 9518$_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC020	XXXX XXXX$_H$	32	FFFF 951C$_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL020	0000 0000$_H$	32	FFFF 9520$_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST020	0000 0000$_H$	32	FFFF 9524$_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS020	0000 0000$_H$	32	FFFF 9528$_H$	0	32

(151/201)

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
DMASS	DTSFSL Transfer Request Clear Register	DTFSC020	0000 0000_H	32	FFFF $952C_H$	0	32
DMASS	DTS Source Address Register	DTSA021	XXXX $XXXX_H$	32	FFFF 9540_H	0	32
DMASS	DTS Destination Address Register	DTDA021	XXXX $XXXX_H$	32	FFFF 9544_H	0	32
DMASS	DTS Transfer Count Register	DTTC021	XXXX $XXXX_H$	32	FFFF 9548_H	0	32
DMASS	DTS Transfer Control Register	DTTCT021	XXXX $XXXX_H$	32	FFFF $954C_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA021	XXXX $XXXX_H$	32	FFFF 9550_H	0	32
DMASS	DTS Reload Destination Address Register	DTRDA021	XXXX $XXXX_H$	32	FFFF 9554_H	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC021	XXXX $XXXX_H$	32	FFFF 9558_H	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC021	XXXX $XXXX_H$	32	FFFF $955C_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL021	0000 0000_H	32	FFFF 9560_H	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST021	0000 0000_H	32	FFFF 9564_H	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS021	0000 0000_H	32	FFFF 9568_H	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC021	0000 0000_H	32	FFFF $956C_H$	0	32
DMASS	DTS Source Address Register	DTSA022	XXXX $XXXX_H$	32	FFFF 9580_H	0	32
DMASS	DTS Destination Address Register	DTDA022	XXXX $XXXX_H$	32	FFFF 9584_H	0	32
DMASS	DTS Transfer Count Register	DTTC022	XXXX $XXXX_H$	32	FFFF 9588_H	0	32
DMASS	DTS Transfer Control Register	DTTCT022	XXXX $XXXX_H$	32	FFFF $958C_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA022	XXXX $XXXX_H$	32	FFFF 9590_H	0	32
DMASS	DTS Reload Destination Address Register	DTRDA022	XXXX $XXXX_H$	32	FFFF 9594_H	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC022	XXXX $XXXX_H$	32	FFFF 9598_H	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC022	XXXX $XXXX_H$	32	FFFF $959C_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL022	0000 0000_H	32	FFFF $95A0_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST022	0000 0000_H	32	FFFF $95A4_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS022	0000 0000_H	32	FFFF $95A8_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC022	0000 0000_H	32	FFFF $95AC_H$	0	32
DMASS	DTS Source Address Register	DTSA023	XXXX $XXXX_H$	32	FFFF $95C0_H$	0	32
DMASS	DTS Destination Address Register	DTDA023	XXXX $XXXX_H$	32	FFFF $95C4_H$	0	32
DMASS	DTS Transfer Count Register	DTTC023	XXXX $XXXX_H$	32	FFFF $95C8_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT023	XXXX $XXXX_H$	32	FFFF $95CC_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA023	XXXX $XXXX_H$	32	FFFF $95D0_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA023	XXXX $XXXX_H$	32	FFFF $95D4_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC023	XXXX $XXXX_H$	32	FFFF $95D8_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC023	XXXX $XXXX_H$	32	FFFF $95DC_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL023	0000 0000_H	32	FFFF $95E0_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST023	0000 0000_H	32	FFFF $95E4_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS023	0000 0000_H	32	FFFF $95E8_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC023	0000 0000_H	32	FFFF $95EC_H$	0	32
DMASS	DTS Source Address Register	DTSA024	XXXX $XXXX_H$	32	FFFF 9600_H	0	32
DMASS	DTS Destination Address Register	DTDA024	XXXX $XXXX_H$	32	FFFF 9604_H	0	32
DMASS	DTS Transfer Count Register	DTTC024	XXXX $XXXX_H$	32	FFFF 9608_H	0	32
DMASS	DTS Transfer Control Register	DTTCT024	XXXX $XXXX_H$	32	FFFF $960C_H$	0	32

(152/201)

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
DMASS	DTS Reload Source Address Register	DTRSA024	XXXX XXXX$_H$	32	FFFF 9610$_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA024	XXXX XXXX$_H$	32	FFFF 9614$_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC024	XXXX XXXX$_H$	32	FFFF 9618$_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC024	XXXX XXXX$_H$	32	FFFF 961C$_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL024	0000 0000$_H$	32	FFFF 9620$_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST024	0000 0000$_H$	32	FFFF 9624$_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS024	0000 0000$_H$	32	FFFF 9628$_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC024	0000 0000$_H$	32	FFFF 962C$_H$	0	32
DMASS	DTS Source Address Register	DTSA025	XXXX XXXX$_H$	32	FFFF 9640$_H$	0	32
DMASS	DTS Destination Address Register	DTDA025	XXXX XXXX$_H$	32	FFFF 9644$_H$	0	32
DMASS	DTS Transfer Count Register	DTTC025	XXXX XXXX$_H$	32	FFFF 9648$_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT025	XXXX XXXX$_H$	32	FFFF 964C$_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA025	XXXX XXXX$_H$	32	FFFF 9650$_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA025	XXXX XXXX$_H$	32	FFFF 9654$_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC025	XXXX XXXX$_H$	32	FFFF 9658$_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC025	XXXX XXXX$_H$	32	FFFF 965C$_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL025	0000 0000$_H$	32	FFFF 9660$_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST025	0000 0000$_H$	32	FFFF 9664$_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS025	0000 0000$_H$	32	FFFF 9668$_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC025	0000 0000$_H$	32	FFFF 966C$_H$	0	32
DMASS	DTS Source Address Register	DTSA026	XXXX XXXX$_H$	32	FFFF 9680$_H$	0	32
DMASS	DTS Destination Address Register	DTDA026	XXXX XXXX$_H$	32	FFFF 9684$_H$	0	32
DMASS	DTS Transfer Count Register	DTTC026	XXXX XXXX$_H$	32	FFFF 9688$_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT026	XXXX XXXX$_H$	32	FFFF 968C$_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA026	XXXX XXXX$_H$	32	FFFF 9690$_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA026	XXXX XXXX$_H$	32	FFFF 9694$_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC026	XXXX XXXX$_H$	32	FFFF 9698$_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC026	XXXX XXXX$_H$	32	FFFF 969C$_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL026	0000 0000$_H$	32	FFFF 96A0$_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST026	0000 0000$_H$	32	FFFF 96A4$_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS026	0000 0000$_H$	32	FFFF 96A8$_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC026	0000 0000$_H$	32	FFFF 96AC$_H$	0	32
DMASS	DTS Source Address Register	DTSA027	XXXX XXXX$_H$	32	FFFF 96C0$_H$	0	32
DMASS	DTS Destination Address Register	DTDA027	XXXX XXXX$_H$	32	FFFF 96C4$_H$	0	32
DMASS	DTS Transfer Count Register	DTTC027	XXXX XXXX$_H$	32	FFFF 96C8$_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT027	XXXX XXXX$_H$	32	FFFF 96CC$_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA027	XXXX XXXX$_H$	32	FFFF 96D0$_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA027	XXXX XXXX$_H$	32	FFFF 96D4$_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC027	XXXX XXXX$_H$	32	FFFF 96D8$_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC027	XXXX XXXX$_H$	32	FFFF 96DC$_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL027	0000 0000$_H$	32	FFFF 96E0$_H$	0	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
DMASS	DTSFSL Transfer Request Status Register	DTFST027	0000 0000_H	32	FFFF $96E4_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS027	0000 0000_H	32	FFFF $96E8_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC027	0000 0000_H	32	FFFF $96EC_H$	0	32
DMASS	DTS Source Address Register	DTSA028	XXXX $XXXX_H$	32	FFFF 9700_H	0	32
DMASS	DTS Destination Address Register	DTDA028	XXXX $XXXX_H$	32	FFFF 9704_H	0	32
DMASS	DTS Transfer Count Register	DTTC028	XXXX $XXXX_H$	32	FFFF 9708_H	0	32
DMASS	DTS Transfer Control Register	DTTCT028	XXXX $XXXX_H$	32	FFFF $970C_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA028	XXXX $XXXX_H$	32	FFFF 9710_H	0	32
DMASS	DTS Reload Destination Address Register	DTRDA028	XXXX $XXXX_H$	32	FFFF 9714_H	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC028	XXXX $XXXX_H$	32	FFFF 9718_H	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC028	XXXX $XXXX_H$	32	FFFF $971C_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL028	0000 0000_H	32	FFFF 9720_H	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST028	0000 0000_H	32	FFFF 9724_H	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS028	0000 0000_H	32	FFFF 9728_H	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC028	0000 0000_H	32	FFFF $972C_H$	0	32
DMASS	DTS Source Address Register	DTSA029	XXXX $XXXX_H$	32	FFFF 9740_H	0	32
DMASS	DTS Destination Address Register	DTDA029	XXXX $XXXX_H$	32	FFFF 9744_H	0	32
DMASS	DTS Transfer Count Register	DTTC029	XXXX $XXXX_H$	32	FFFF 9748_H	0	32
DMASS	DTS Transfer Control Register	DTTCT029	XXXX $XXXX_H$	32	FFFF $974C_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA029	XXXX $XXXX_H$	32	FFFF 9750_H	0	32
DMASS	DTS Reload Destination Address Register	DTRDA029	XXXX $XXXX_H$	32	FFFF 9754_H	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC029	XXXX $XXXX_H$	32	FFFF 9758_H	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC029	XXXX $XXXX_H$	32	FFFF $975C_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL029	0000 0000_H	32	FFFF 9760_H	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST029	0000 0000_H	32	FFFF 9764_H	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS029	0000 0000_H	32	FFFF 9768_H	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC029	0000 0000_H	32	FFFF $976C_H$	0	32
DMASS	DTS Source Address Register	DTSA030	XXXX $XXXX_H$	32	FFFF 9780_H	0	32
DMASS	DTS Destination Address Register	DTDA030	XXXX $XXXX_H$	32	FFFF 9784_H	0	32
DMASS	DTS Transfer Count Register	DTTC030	XXXX $XXXX_H$	32	FFFF 9788_H	0	32
DMASS	DTS Transfer Control Register	DTTCT030	XXXX $XXXX_H$	32	FFFF $978C_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA030	XXXX $XXXX_H$	32	FFFF 9790_H	0	32
DMASS	DTS Reload Destination Address Register	DTRDA030	XXXX $XXXX_H$	32	FFFF 9794_H	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC030	XXXX $XXXX_H$	32	FFFF 9798_H	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC030	XXXX $XXXX_H$	32	FFFF $979C_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL030	0000 0000_H	32	FFFF $97A0_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST030	0000 0000_H	32	FFFF $97A4_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS030	0000 0000_H	32	FFFF $97A8_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC030	0000 0000_H	32	FFFF $97AC_H$	0	32
DMASS	DTS Source Address Register	DTSA031	XXXX $XXXX_H$	32	FFFF $97C0_H$	0	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
DMASS	DTS Destination Address Register	DTDA031	XXXX XXXX$_H$	32	FFFF 97C4$_H$	0	32
DMASS	DTS Transfer Count Register	DTTC031	XXXX XXXX$_H$	32	FFFF 97C8$_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT031	XXXX XXXX$_H$	32	FFFF 97CC$_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA031	XXXX XXXX$_H$	32	FFFF 97D0$_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA031	XXXX XXXX$_H$	32	FFFF 97D4$_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC031	XXXX XXXX$_H$	32	FFFF 97D8$_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC031	XXXX XXXX$_H$	32	FFFF 97DC$_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL031	0000 0000$_H$	32	FFFF 97E0$_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST031	0000 0000$_H$	32	FFFF 97E4$_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS031	0000 0000$_H$	32	FFFF 97E8$_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC031	0000 0000$_H$	32	FFFF 97EC$_H$	0	32
DMASS	DTS Source Address Register	DTSA032	XXXX XXXX$_H$	32	FFFF 9800$_H$	0	32
DMASS	DTS Destination Address Register	DTDA032	XXXX XXXX$_H$	32	FFFF 9804$_H$	0	32
DMASS	DTS Transfer Count Register	DTTC032	XXXX XXXX$_H$	32	FFFF 9808$_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT032	XXXX XXXX$_H$	32	FFFF 980C$_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA032	XXXX XXXX$_H$	32	FFFF 9810$_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA032	XXXX XXXX$_H$	32	FFFF 9814$_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC032	XXXX XXXX$_H$	32	FFFF 9818$_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC032	XXXX XXXX$_H$	32	FFFF 981C$_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL032	0000 0000$_H$	32	FFFF 9820$_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST032	0000 0000$_H$	32	FFFF 9824$_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS032	0000 0000$_H$	32	FFFF 9828$_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC032	0000 0000$_H$	32	FFFF 982C$_H$	0	32
DMASS	DTS Source Address Register	DTSA033	XXXX XXXX$_H$	32	FFFF 9840$_H$	0	32
DMASS	DTS Destination Address Register	DTDA033	XXXX XXXX$_H$	32	FFFF 9844$_H$	0	32
DMASS	DTS Transfer Count Register	DTTC033	XXXX XXXX$_H$	32	FFFF 9848$_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT033	XXXX XXXX$_H$	32	FFFF 984C$_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA033	XXXX XXXX$_H$	32	FFFF 9850$_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA033	XXXX XXXX$_H$	32	FFFF 9854$_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC033	XXXX XXXX$_H$	32	FFFF 9858$_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC033	XXXX XXXX$_H$	32	FFFF 985C$_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL033	0000 0000$_H$	32	FFFF 9860$_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST033	0000 0000$_H$	32	FFFF 9864$_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS033	0000 0000$_H$	32	FFFF 9868$_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC033	0000 0000$_H$	32	FFFF 986C$_H$	0	32
DMASS	DTS Source Address Register	DTSA034	XXXX XXXX$_H$	32	FFFF 9880$_H$	0	32
DMASS	DTS Destination Address Register	DTDA034	XXXX XXXX$_H$	32	FFFF 9884$_H$	0	32
DMASS	DTS Transfer Count Register	DTTC034	XXXX XXXX$_H$	32	FFFF 9888$_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT034	XXXX XXXX$_H$	32	FFFF 988C$_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA034	XXXX XXXX$_H$	32	FFFF 9890$_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA034	XXXX XXXX$_H$	32	FFFF 9894$_H$	0	32

(155/201)

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
DMASS	DTS Reload Transfer Count Register	DTRTC034	XXXX $XXXX_H$	32	FFFF 9898_H	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC034	XXXX $XXXX_H$	32	FFFF $989C_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL034	0000 0000_H	32	FFFF $98A0_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST034	0000 0000_H	32	FFFF $98A4_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS034	0000 0000_H	32	FFFF $98A8_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC034	0000 0000_H	32	FFFF $98AC_H$	0	32
DMASS	DTS Source Address Register	DTSA035	XXXX $XXXX_H$	32	FFFF $98C0_H$	0	32
DMASS	DTS Destination Address Register	DTDA035	XXXX $XXXX_H$	32	FFFF $98C4_H$	0	32
DMASS	DTS Transfer Count Register	DTTC035	XXXX $XXXX_H$	32	FFFF $98C8_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT035	XXXX $XXXX_H$	32	FFFF $98CC_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA035	XXXX $XXXX_H$	32	FFFF $98D0_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA035	XXXX $XXXX_H$	32	FFFF $98D4_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC035	XXXX $XXXX_H$	32	FFFF $98D8_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC035	XXXX $XXXX_H$	32	FFFF $98DC_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL035	0000 0000_H	32	FFFF $98E0_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST035	0000 0000_H	32	FFFF $98E4_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS035	0000 0000_H	32	FFFF $98E8_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC035	0000 0000_H	32	FFFF $98EC_H$	0	32
DMASS	DTS Source Address Register	DTSA036	XXXX $XXXX_H$	32	FFFF 9900_H	0	32
DMASS	DTS Destination Address Register	DTDA036	XXXX $XXXX_H$	32	FFFF 9904_H	0	32
DMASS	DTS Transfer Count Register	DTTC036	XXXX $XXXX_H$	32	FFFF 9908_H	0	32
DMASS	DTS Transfer Control Register	DTTCT036	XXXX $XXXX_H$	32	FFFF $990C_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA036	XXXX $XXXX_H$	32	FFFF 9910_H	0	32
DMASS	DTS Reload Destination Address Register	DTRDA036	XXXX $XXXX_H$	32	FFFF 9914_H	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC036	XXXX $XXXX_H$	32	FFFF 9918_H	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC036	XXXX $XXXX_H$	32	FFFF $991C_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL036	0000 0000_H	32	FFFF 9920_H	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST036	0000 0000_H	32	FFFF 9924_H	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS036	0000 0000_H	32	FFFF 9928_H	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC036	0000 0000_H	32	FFFF $992C_H$	0	32
DMASS	DTS Source Address Register	DTSA037	XXXX $XXXX_H$	32	FFFF 9940_H	0	32
DMASS	DTS Destination Address Register	DTDA037	XXXX $XXXX_H$	32	FFFF 9944_H	0	32
DMASS	DTS Transfer Count Register	DTTC037	XXXX $XXXX_H$	32	FFFF 9948_H	0	32
DMASS	DTS Transfer Control Register	DTTCT037	XXXX $XXXX_H$	32	FFFF $994C_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA037	XXXX $XXXX_H$	32	FFFF 9950_H	0	32
DMASS	DTS Reload Destination Address Register	DTRDA037	XXXX $XXXX_H$	32	FFFF 9954_H	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC037	XXXX $XXXX_H$	32	FFFF 9958_H	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC037	XXXX $XXXX_H$	32	FFFF $995C_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL037	0000 0000_H	32	FFFF 9960_H	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST037	0000 0000_H	32	FFFF 9964_H	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS037	0000 0000_H	32	FFFF 9968_H	0	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
DMASS	DTSFSL Transfer Request Clear Register	DTFSC037	0000 0000_H	32	FFFF $996C_H$	0	32
DMASS	DTS Source Address Register	DTSA038	XXXX $XXXX_H$	32	FFFF 9980_H	0	32
DMASS	DTS Destination Address Register	DTDA038	XXXX $XXXX_H$	32	FFFF 9984_H	0	32
DMASS	DTS Transfer Count Register	DTTC038	XXXX $XXXX_H$	32	FFFF 9988_H	0	32
DMASS	DTS Transfer Control Register	DTTCT038	XXXX $XXXX_H$	32	FFFF $998C_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA038	XXXX $XXXX_H$	32	FFFF 9990_H	0	32
DMASS	DTS Reload Destination Address Register	DTRDA038	XXXX $XXXX_H$	32	FFFF 9994_H	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC038	XXXX $XXXX_H$	32	FFFF 9998_H	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC038	XXXX $XXXX_H$	32	FFFF $999C_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL038	0000 0000_H	32	FFFF $99A0_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST038	0000 0000_H	32	FFFF $99A4_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS038	0000 0000_H	32	FFFF $99A8_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC038	0000 0000_H	32	FFFF $99AC_H$	0	32
DMASS	DTS Source Address Register	DTSA039	XXXX $XXXX_H$	32	FFFF $99C0_H$	0	32
DMASS	DTS Destination Address Register	DTDA039	XXXX $XXXX_H$	32	FFFF $99C4_H$	0	32
DMASS	DTS Transfer Count Register	DTTC039	XXXX $XXXX_H$	32	FFFF $99C8_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT039	XXXX $XXXX_H$	32	FFFF $99CC_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA039	XXXX $XXXX_H$	32	FFFF $99D0_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA039	XXXX $XXXX_H$	32	FFFF $99D4_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC039	XXXX $XXXX_H$	32	FFFF $99D8_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC039	XXXX $XXXX_H$	32	FFFF $99DC_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL039	0000 0000_H	32	FFFF $99E0_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST039	0000 0000_H	32	FFFF $99E4_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS039	0000 0000_H	32	FFFF $99E8_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC039	0000 0000_H	32	FFFF $99EC_H$	0	32
DMASS	DTS Source Address Register	DTSA040	XXXX $XXXX_H$	32	FFFF $9A00_H$	0	32
DMASS	DTS Destination Address Register	DTDA040	XXXX $XXXX_H$	32	FFFF $9A04_H$	0	32
DMASS	DTS Transfer Count Register	DTTC040	XXXX $XXXX_H$	32	FFFF $9A08_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT040	XXXX $XXXX_H$	32	FFFF $9A0C_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA040	XXXX $XXXX_H$	32	FFFF $9A10_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA040	XXXX $XXXX_H$	32	FFFF $9A14_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC040	XXXX $XXXX_H$	32	FFFF $9A18_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC040	XXXX $XXXX_H$	32	FFFF $9A1C_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL040	0000 0000_H	32	FFFF $9A20_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST040	0000 0000_H	32	FFFF $9A24_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS040	0000 0000_H	32	FFFF $9A28_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC040	0000 0000_H	32	FFFF $9A2C_H$	0	32
DMASS	DTS Source Address Register	DTSA041	XXXX $XXXX_H$	32	FFFF $9A40_H$	0	32
DMASS	DTS Destination Address Register	DTDA041	XXXX $XXXX_H$	32	FFFF $9A44_H$	0	32
DMASS	DTS Transfer Count Register	DTTC041	XXXX $XXXX_H$	32	FFFF $9A48_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT041	XXXX $XXXX_H$	32	FFFF $9A4C_H$	0	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
DMASS	DTS Reload Source Address Register	DTRSA041	XXXX XXXX$_H$	32	FFFF 9A50$_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA041	XXXX XXXX$_H$	32	FFFF 9A54$_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC041	XXXX XXXX$_H$	32	FFFF 9A58$_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC041	XXXX XXXX$_H$	32	FFFF 9A5C$_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL041	0000 0000$_H$	32	FFFF 9A60$_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST041	0000 0000$_H$	32	FFFF 9A64$_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS041	0000 0000$_H$	32	FFFF 9A68$_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC041	0000 0000$_H$	32	FFFF 9A6C$_H$	0	32
DMASS	DTS Source Address Register	DTSA042	XXXX XXXX$_H$	32	FFFF 9A80$_H$	0	32
DMASS	DTS Destination Address Register	DTDA042	XXXX XXXX$_H$	32	FFFF 9A84$_H$	0	32
DMASS	DTS Transfer Count Register	DTTC042	XXXX XXXX$_H$	32	FFFF 9A88$_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT042	XXXX XXXX$_H$	32	FFFF 9A8C$_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA042	XXXX XXXX$_H$	32	FFFF 9A90$_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA042	XXXX XXXX$_H$	32	FFFF 9A94$_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC042	XXXX XXXX$_H$	32	FFFF 9A98$_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC042	XXXX XXXX$_H$	32	FFFF 9A9C$_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL042	0000 0000$_H$	32	FFFF 9AA0$_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST042	0000 0000$_H$	32	FFFF 9AA4$_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS042	0000 0000$_H$	32	FFFF 9AA8$_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC042	0000 0000$_H$	32	FFFF 9AAC$_H$	0	32
DMASS	DTS Source Address Register	DTSA043	XXXX XXXX$_H$	32	FFFF 9AC0$_H$	0	32
DMASS	DTS Destination Address Register	DTDA043	XXXX XXXX$_H$	32	FFFF 9AC4$_H$	0	32
DMASS	DTS Transfer Count Register	DTTC043	XXXX XXXX$_H$	32	FFFF 9AC8$_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT043	XXXX XXXX$_H$	32	FFFF 9ACC$_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA043	XXXX XXXX$_H$	32	FFFF 9AD0$_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA043	XXXX XXXX$_H$	32	FFFF 9AD4$_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC043	XXXX XXXX$_H$	32	FFFF 9AD8$_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC043	XXXX XXXX$_H$	32	FFFF 9ADC$_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL043	0000 0000$_H$	32	FFFF 9AE0$_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST043	0000 0000$_H$	32	FFFF 9AE4$_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS043	0000 0000$_H$	32	FFFF 9AE8$_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC043	0000 0000$_H$	32	FFFF 9AEC$_H$	0	32
DMASS	DTS Source Address Register	DTSA044	XXXX XXXX$_H$	32	FFFF 9B00$_H$	0	32
DMASS	DTS Destination Address Register	DTDA044	XXXX XXXX$_H$	32	FFFF 9B04$_H$	0	32
DMASS	DTS Transfer Count Register	DTTC044	XXXX XXXX$_H$	32	FFFF 9B08$_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT044	XXXX XXXX$_H$	32	FFFF 9B0C$_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA044	XXXX XXXX$_H$	32	FFFF 9B10$_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA044	XXXX XXXX$_H$	32	FFFF 9B14$_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC044	XXXX XXXX$_H$	32	FFFF 9B18$_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC044	XXXX XXXX$_H$	32	FFFF 9B1C$_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL044	0000 0000$_H$	32	FFFF 9B20$_H$	0	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
DMASS	DTSFSL Transfer Request Status Register	DTFST044	0000 0000_H	32	FFFF $9B24_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS044	0000 0000_H	32	FFFF $9B28_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC044	0000 0000_H	32	FFFF $9B2C_H$	0	32
DMASS	DTS Source Address Register	DTSA045	XXXX $XXXX_H$	32	FFFF $9B40_H$	0	32
DMASS	DTS Destination Address Register	DTDA045	XXXX $XXXX_H$	32	FFFF $9B44_H$	0	32
DMASS	DTS Transfer Count Register	DTTC045	XXXX $XXXX_H$	32	FFFF $9B48_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT045	XXXX $XXXX_H$	32	FFFF $9B4C_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA045	XXXX $XXXX_H$	32	FFFF $9B50_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA045	XXXX $XXXX_H$	32	FFFF $9B54_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC045	XXXX $XXXX_H$	32	FFFF $9B58_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC045	XXXX $XXXX_H$	32	FFFF $9B5C_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL045	0000 0000_H	32	FFFF $9B60_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST045	0000 0000_H	32	FFFF $9B64_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS045	0000 0000_H	32	FFFF $9B68_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC045	0000 0000_H	32	FFFF $9B6C_H$	0	32
DMASS	DTS Source Address Register	DTSA046	XXXX $XXXX_H$	32	FFFF $9B80_H$	0	32
DMASS	DTS Destination Address Register	DTDA046	XXXX $XXXX_H$	32	FFFF $9B84_H$	0	32
DMASS	DTS Transfer Count Register	DTTC046	XXXX $XXXX_H$	32	FFFF $9B88_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT046	XXXX $XXXX_H$	32	FFFF $9B8C_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA046	XXXX $XXXX_H$	32	FFFF $9B90_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA046	XXXX $XXXX_H$	32	FFFF $9B94_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC046	XXXX $XXXX_H$	32	FFFF $9B98_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC046	XXXX $XXXX_H$	32	FFFF $9B9C_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL046	0000 0000_H	32	FFFF $9BA0_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST046	0000 0000_H	32	FFFF $9BA4_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS046	0000 0000_H	32	FFFF $9BA8_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC046	0000 0000_H	32	FFFF $9BAC_H$	0	32
DMASS	DTS Source Address Register	DTSA047	XXXX $XXXX_H$	32	FFFF $9BC0_H$	0	32
DMASS	DTS Destination Address Register	DTDA047	XXXX $XXXX_H$	32	FFFF $9BC4_H$	0	32
DMASS	DTS Transfer Count Register	DTTC047	XXXX $XXXX_H$	32	FFFF $9BC8_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT047	XXXX $XXXX_H$	32	FFFF $9BCC_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA047	XXXX $XXXX_H$	32	FFFF $9BD0_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA047	XXXX $XXXX_H$	32	FFFF $9BD4_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC047	XXXX $XXXX_H$	32	FFFF $9BD8_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC047	XXXX $XXXX_H$	32	FFFF $9BDC_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL047	0000 0000_H	32	FFFF $9BE0_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST047	0000 0000_H	32	FFFF $9BE4_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS047	0000 0000_H	32	FFFF $9BE8_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC047	0000 0000_H	32	FFFF $9BEC_H$	0	32
DMASS	DTS Source Address Register	DTSA048	XXXX $XXXX_H$	32	FFFF $9C00_H$	0	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
DMASS	DTS Destination Address Register	DTDA048	XXXX XXXX$_H$	32	FFFF 9C04$_H$	0	32
DMASS	DTS Transfer Count Register	DTTC048	XXXX XXXX$_H$	32	FFFF 9C08$_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT048	XXXX XXXX$_H$	32	FFFF 9C0C$_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA048	XXXX XXXX$_H$	32	FFFF 9C10$_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA048	XXXX XXXX$_H$	32	FFFF 9C14$_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC048	XXXX XXXX$_H$	32	FFFF 9C18$_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC048	XXXX XXXX$_H$	32	FFFF 9C1C$_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL048	0000 0000$_H$	32	FFFF 9C20$_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST048	0000 0000$_H$	32	FFFF 9C24$_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS048	0000 0000$_H$	32	FFFF 9C28$_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC048	0000 0000$_H$	32	FFFF 9C2C$_H$	0	32
DMASS	DTS Source Address Register	DTSA049	XXXX XXXX$_H$	32	FFFF 9C40$_H$	0	32
DMASS	DTS Destination Address Register	DTDA049	XXXX XXXX$_H$	32	FFFF 9C44$_H$	0	32
DMASS	DTS Transfer Count Register	DTTC049	XXXX XXXX$_H$	32	FFFF 9C48$_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT049	XXXX XXXX$_H$	32	FFFF 9C4C$_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA049	XXXX XXXX$_H$	32	FFFF 9C50$_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA049	XXXX XXXX$_H$	32	FFFF 9C54$_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC049	XXXX XXXX$_H$	32	FFFF 9C58$_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC049	XXXX XXXX$_H$	32	FFFF 9C5C$_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL049	0000 0000$_H$	32	FFFF 9C60$_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST049	0000 0000$_H$	32	FFFF 9C64$_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS049	0000 0000$_H$	32	FFFF 9C68$_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC049	0000 0000$_H$	32	FFFF 9C6C$_H$	0	32
DMASS	DTS Source Address Register	DTSA050	XXXX XXXX$_H$	32	FFFF 9C80$_H$	0	32
DMASS	DTS Destination Address Register	DTDA050	XXXX XXXX$_H$	32	FFFF 9C84$_H$	0	32
DMASS	DTS Transfer Count Register	DTTC050	XXXX XXXX$_H$	32	FFFF 9C88$_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT050	XXXX XXXX$_H$	32	FFFF 9C8C$_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA050	XXXX XXXX$_H$	32	FFFF 9C90$_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA050	XXXX XXXX$_H$	32	FFFF 9C94$_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC050	XXXX XXXX$_H$	32	FFFF 9C98$_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC050	XXXX XXXX$_H$	32	FFFF 9C9C$_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL050	0000 0000$_H$	32	FFFF 9CA0$_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST050	0000 0000$_H$	32	FFFF 9CA4$_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS050	0000 0000$_H$	32	FFFF 9CA8$_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC050	0000 0000$_H$	32	FFFF 9CAC$_H$	0	32
DMASS	DTS Source Address Register	DTSA051	XXXX XXXX$_H$	32	FFFF 9CC0$_H$	0	32
DMASS	DTS Destination Address Register	DTDA051	XXXX XXXX$_H$	32	FFFF 9CC4$_H$	0	32
DMASS	DTS Transfer Count Register	DTTC051	XXXX XXXX$_H$	32	FFFF 9CC8$_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT051	XXXX XXXX$_H$	32	FFFF 9CCC$_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA051	XXXX XXXX$_H$	32	FFFF 9CD0$_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA051	XXXX XXXX$_H$	32	FFFF 9CD4$_H$	0	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
DMASS	DTS Reload Transfer Count Register	DTRTC051	XXXX $XXXX_H$	32	FFFF $9CD8_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC051	XXXX $XXXX_H$	32	FFFF $9CDC_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL051	0000 0000_H	32	FFFF $9CE0_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST051	0000 0000_H	32	FFFF $9CE4_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS051	0000 0000_H	32	FFFF $9CE8_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC051	0000 0000_H	32	FFFF $9CEC_H$	0	32
DMASS	DTS Source Address Register	DTSA052	XXXX $XXXX_H$	32	FFFF $9D00_H$	0	32
DMASS	DTS Destination Address Register	DTDA052	XXXX $XXXX_H$	32	FFFF $9D04_H$	0	32
DMASS	DTS Transfer Count Register	DTTC052	XXXX $XXXX_H$	32	FFFF $9D08_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT052	XXXX $XXXX_H$	32	FFFF $9D0C_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA052	XXXX $XXXX_H$	32	FFFF $9D10_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA052	XXXX $XXXX_H$	32	FFFF $9D14_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC052	XXXX $XXXX_H$	32	FFFF $9D18_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC052	XXXX $XXXX_H$	32	FFFF $9D1C_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL052	0000 0000_H	32	FFFF $9D20_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST052	0000 0000_H	32	FFFF $9D24_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS052	0000 0000_H	32	FFFF $9D28_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC052	0000 0000_H	32	FFFF $9D2C_H$	0	32
DMASS	DTS Source Address Register	DTSA053	XXXX $XXXX_H$	32	FFFF $9D40_H$	0	32
DMASS	DTS Destination Address Register	DTDA053	XXXX $XXXX_H$	32	FFFF $9D44_H$	0	32
DMASS	DTS Transfer Count Register	DTTC053	XXXX $XXXX_H$	32	FFFF $9D48_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT053	XXXX $XXXX_H$	32	FFFF $9D4C_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA053	XXXX $XXXX_H$	32	FFFF $9D50_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA053	XXXX $XXXX_H$	32	FFFF $9D54_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC053	XXXX $XXXX_H$	32	FFFF $9D58_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC053	XXXX $XXXX_H$	32	FFFF $9D5C_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL053	0000 0000_H	32	FFFF $9D60_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST053	0000 0000_H	32	FFFF $9D64_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS053	0000 0000_H	32	FFFF $9D68_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC053	0000 0000_H	32	FFFF $9D6C_H$	0	32
DMASS	DTS Source Address Register	DTSA054	XXXX $XXXX_H$	32	FFFF $9D80_H$	0	32
DMASS	DTS Destination Address Register	DTDA054	XXXX $XXXX_H$	32	FFFF $9D84_H$	0	32
DMASS	DTS Transfer Count Register	DTTC054	XXXX $XXXX_H$	32	FFFF $9D88_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT054	XXXX $XXXX_H$	32	FFFF $9D8C_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA054	XXXX $XXXX_H$	32	FFFF $9D90_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA054	XXXX $XXXX_H$	32	FFFF $9D94_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC054	XXXX $XXXX_H$	32	FFFF $9D98_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC054	XXXX $XXXX_H$	32	FFFF $9D9C_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL054	0000 0000_H	32	FFFF $9DA0_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST054	0000 0000_H	32	FFFF $9DA4_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS054	0000 0000_H	32	FFFF $9DA8_H$	0	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
DMASS	DTSFSL Transfer Request Clear Register	DTFSC054	0000 0000_H	32	FFFF $9DAC_H$	0	32
DMASS	DTS ??????????	DTSA055	XXXX $XXXX_H$	32	FFFF $9DC0_H$	0	32
DMASS	DTS ?????????????????	DTDA055	XXXX $XXXX_H$	32	FFFF $9DC4_H$	0	32
DMASS	DTS Source Address Register	DTTC055	XXXX $XXXX_H$	32	FFFF $9DC8_H$	0	32
DMASS	DTS Destination Address Register	DTTCT055	XXXX $XXXX_H$	32	FFFF $9DCC_H$	0	32
DMASS	DTS Transfer Count Register	DTRSA055	XXXX $XXXX_H$	32	FFFF $9DD0_H$	0	32
DMASS	DTS Transfer Control Register	DTRDA055	XXXX $XXXX_H$	32	FFFF $9DD4_H$	0	32
DMASS	DTS Reload Source Address Register	DTRTC055	XXXX $XXXX_H$	32	FFFF $9DD8_H$	0	32
DMASS	DTS Reload Destination Address Register	DTTCC055	XXXX $XXXX_H$	32	FFFF $9DDC_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTFSL055	0000 0000_H	32	FFFF $9DE0_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTFST055	0000 0000_H	32	FFFF $9DE4_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSS055	0000 0000_H	32	FFFF $9DE8_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFSC055	0000 0000_H	32	FFFF $9DEC_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTSA056	XXXX $XXXX_H$	32	FFFF $9E00_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTDA056	XXXX $XXXX_H$	32	FFFF $9E04_H$	0	32
DMASS	DTS Source Address Register	DTTC056	XXXX $XXXX_H$	32	FFFF $9E08_H$	0	32
DMASS	DTS Destination Address Register	DTTCT056	XXXX $XXXX_H$	32	FFFF $9E0C_H$	0	32
DMASS	DTS Transfer Count Register	DTRSA056	XXXX $XXXX_H$	32	FFFF $9E10_H$	0	32
DMASS	DTS Transfer Control Register	DTRDA056	XXXX $XXXX_H$	32	FFFF $9E14_H$	0	32
DMASS	DTS Reload Source Address Register	DTRTC056	XXXX $XXXX_H$	32	FFFF $9E18_H$	0	32
DMASS	DTS Reload Destination Address Register	DTTCC056	XXXX $XXXX_H$	32	FFFF $9E1C_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTFSL056	0000 0000_H	32	FFFF $9E20_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTFST056	0000 0000_H	32	FFFF $9E24_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSS056	0000 0000_H	32	FFFF $9E28_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFSC056	0000 0000_H	32	FFFF $9E2C_H$	0	32
DMASS	DTS Source Address Register	DTSA057	XXXX $XXXX_H$	32	FFFF $9E40_H$	0	32
DMASS	DTS Destination Address Register	DTDA057	XXXX $XXXX_H$	32	FFFF $9E44_H$	0	32
DMASS	DTS Transfer Count Register	DTTC057	XXXX $XXXX_H$	32	FFFF $9E48_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT057	XXXX $XXXX_H$	32	FFFF $9E4C_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA057	XXXX $XXXX_H$	32	FFFF $9E50_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA057	XXXX $XXXX_H$	32	FFFF $9E54_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC057	XXXX $XXXX_H$	32	FFFF $9E58_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC057	XXXX $XXXX_H$	32	FFFF $9E5C_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL057	0000 0000_H	32	FFFF $9E60_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST057	0000 0000_H	32	FFFF $9E64_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS057	0000 0000_H	32	FFFF $9E68_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC057	0000 0000_H	32	FFFF $9E6C_H$	0	32
DMASS	DTS Source Address Register	DTSA058	XXXX $XXXX_H$	32	FFFF $9E80_H$	0	32
DMASS	DTS Destination Address Register	DTDA058	XXXX $XXXX_H$	32	FFFF $9E84_H$	0	32
DMASS	DTS Transfer Count Register	DTTC058	XXXX $XXXX_H$	32	FFFF $9E88_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT058	XXXX $XXXX_H$	32	FFFF $9E8C_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA058	XXXX $XXXX_H$	32	FFFF $9E90_H$	0	32

(162/201)

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
DMASS	DTS Reload Destination Address Register	DTRDA058	XXXX $XXXX_H$	32	FFFF $9E94_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC058	XXXX $XXXX_H$	32	FFFF $9E98_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC058	XXXX $XXXX_H$	32	FFFF $9E9C_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL058	0000 0000_H	32	FFFF $9EA0_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST058	0000 0000_H	32	FFFF $9EA4_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS058	0000 0000_H	32	FFFF $9EA8_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC058	0000 0000_H	32	FFFF $9EAC_H$	0	32
DMASS	DTS Source Address Register	DTSA059	XXXX $XXXX_H$	32	FFFF $9EC0_H$	0	32
DMASS	DTS Destination Address Register	DTDA059	XXXX $XXXX_H$	32	FFFF $9EC4_H$	0	32
DMASS	DTS Transfer Count Register	DTTC059	XXXX $XXXX_H$	32	FFFF $9EC8_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT059	XXXX $XXXX_H$	32	FFFF $9ECC_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA059	XXXX $XXXX_H$	32	FFFF $9ED0_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA059	XXXX $XXXX_H$	32	FFFF $9ED4_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC059	XXXX $XXXX_H$	32	FFFF $9ED8_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC059	XXXX $XXXX_H$	32	FFFF $9EDC_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL059	0000 0000_H	32	FFFF $9EE0_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST059	0000 0000_H	32	FFFF $9EE4_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS059	0000 0000_H	32	FFFF $9EE8_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC059	0000 0000_H	32	FFFF $9EEC_H$	0	32
DMASS	DTS Source Address Register	DTSA060	XXXX $XXXX_H$	32	FFFF $9F00_H$	0	32
DMASS	DTS Destination Address Register	DTDA060	XXXX $XXXX_H$	32	FFFF $9F04_H$	0	32
DMASS	DTS Transfer Count Register	DTTC060	XXXX $XXXX_H$	32	FFFF $9F08_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT060	XXXX $XXXX_H$	32	FFFF $9F0C_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA060	XXXX $XXXX_H$	32	FFFF $9F10_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA060	XXXX $XXXX_H$	32	FFFF $9F14_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC060	XXXX $XXXX_H$	32	FFFF $9F18_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC060	XXXX $XXXX_H$	32	FFFF $9F1C_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL060	0000 0000_H	32	FFFF $9F20_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST060	0000 0000_H	32	FFFF $9F24_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS060	0000 0000_H	32	FFFF $9F28_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC060	0000 0000_H	32	FFFF $9F2C_H$	0	32
DMASS	DTS Source Address Register	DTSA061	XXXX $XXXX_H$	32	FFFF $9F40_H$	0	32
DMASS	DTS Destination Address Register	DTDA061	XXXX $XXXX_H$	32	FFFF $9F44_H$	0	32
DMASS	DTS Transfer Count Register	DTTC061	XXXX $XXXX_H$	32	FFFF $9F48_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT061	XXXX $XXXX_H$	32	FFFF $9F4C_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA061	XXXX $XXXX_H$	32	FFFF $9F50_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA061	XXXX $XXXX_H$	32	FFFF $9F54_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC061	XXXX $XXXX_H$	32	FFFF $9F58_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC061	XXXX $XXXX_H$	32	FFFF $9F5C_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL061	0000 0000_H	32	FFFF $9F60_H$	0	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
DMASS	DTSFSL Transfer Request Status Register	DTFST061	0000 0000_H	32	FFFF $9F64_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS061	0000 0000_H	32	FFFF $9F68_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC061	0000 0000_H	32	FFFF $9F6C_H$	0	32
DMASS	DTS Source Address Register	DTSA062	XXXX $XXXX_H$	32	FFFF $9F80_H$	0	32
DMASS	DTS Destination Address Register	DTDA062	XXXX $XXXX_H$	32	FFFF $9F84_H$	0	32
DMASS	DTS Transfer Count Register	DTTC062	XXXX $XXXX_H$	32	FFFF $9F88_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT062	XXXX $XXXX_H$	32	FFFF $9F8C_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA062	XXXX $XXXX_H$	32	FFFF $9F90_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA062	XXXX $XXXX_H$	32	FFFF $9F94_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC062	XXXX $XXXX_H$	32	FFFF $9F98_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC062	XXXX $XXXX_H$	32	FFFF $9F9C_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL062	0000 0000_H	32	FFFF $9FA0_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST062	0000 0000_H	32	FFFF $9FA4_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS062	0000 0000_H	32	FFFF $9FA8_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC062	0000 0000_H	32	FFFF $9FAC_H$	0	32
DMASS	DTS Source Address Register	DTSA063	XXXX $XXXX_H$	32	FFFF $9FC0_H$	0	32
DMASS	DTS Destination Address Register	DTDA063	XXXX $XXXX_H$	32	FFFF $9FC4_H$	0	32
DMASS	DTS Transfer Count Register	DTTC063	XXXX $XXXX_H$	32	FFFF $9FC8_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT063	XXXX $XXXX_H$	32	FFFF $9FCC_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA063	XXXX $XXXX_H$	32	FFFF $9FD0_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA063	XXXX $XXXX_H$	32	FFFF $9FD4_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC063	XXXX $XXXX_H$	32	FFFF $9FD8_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC063	XXXX $XXXX_H$	32	FFFF $9FDC_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL063	0000 0000_H	32	FFFF $9FE0_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST063	0000 0000_H	32	FFFF $9FE4_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS063	0000 0000_H	32	FFFF $9FE8_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC063	0000 0000_H	32	FFFF $9FEC_H$	0	32
DMASS	DTS Source Address Register	DTSA064	XXXX $XXXX_H$	32	FFFF $A000_H$	0	32
DMASS	DTS Destination Address Register	DTDA064	XXXX $XXXX_H$	32	FFFF $A004_H$	0	32
DMASS	DTS Transfer Count Register	DTTC064	XXXX $XXXX_H$	32	FFFF $A008_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT064	XXXX $XXXX_H$	32	FFFF $A00C_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA064	XXXX $XXXX_H$	32	FFFF $A010_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA064	XXXX $XXXX_H$	32	FFFF $A014_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC064	XXXX $XXXX_H$	32	FFFF $A018_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC064	XXXX $XXXX_H$	32	FFFF $A01C_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL064	0000 0000_H	32	FFFF $A020_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST064	0000 0000_H	32	FFFF $A024_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS064	0000 0000_H	32	FFFF $A028_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC064	0000 0000_H	32	FFFF $A02C_H$	0	32
DMASS	DTS Source Address Register	DTSA065	XXXX $XXXX_H$	32	FFFF $A040_H$	0	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
DMASS	DTS Destination Address Register	DTDA065	XXXX XXXX$_H$	32	FFFF A044$_H$	0	32
DMASS	DTS Transfer Count Register	DTTC065	XXXX XXXX$_H$	32	FFFF A048$_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT065	XXXX XXXX$_H$	32	FFFF A04C$_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA065	XXXX XXXX$_H$	32	FFFF A050$_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA065	XXXX XXXX$_H$	32	FFFF A054$_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC065	XXXX XXXX$_H$	32	FFFF A058$_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC065	XXXX XXXX$_H$	32	FFFF A05C$_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL065	0000 0000$_H$	32	FFFF A060$_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST065	0000 0000$_H$	32	FFFF A064$_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS065	0000 0000$_H$	32	FFFF A068$_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC065	0000 0000$_H$	32	FFFF A06C$_H$	0	32
DMASS	DTS Source Address Register	DTSA066	XXXX XXXX$_H$	32	FFFF A080$_H$	0	32
DMASS	DTS Destination Address Register	DTDA066	XXXX XXXX$_H$	32	FFFF A084$_H$	0	32
DMASS	DTS Transfer Count Register	DTTC066	XXXX XXXX$_H$	32	FFFF A088$_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT066	XXXX XXXX$_H$	32	FFFF A08C$_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA066	XXXX XXXX$_H$	32	FFFF A090$_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA066	XXXX XXXX$_H$	32	FFFF A094$_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC066	XXXX XXXX$_H$	32	FFFF A098$_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC066	XXXX XXXX$_H$	32	FFFF A09C$_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL066	0000 0000$_H$	32	FFFF A0A0$_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST066	0000 0000$_H$	32	FFFF A0A4$_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS066	0000 0000$_H$	32	FFFF A0A8$_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC066	0000 0000$_H$	32	FFFF A0AC$_H$	0	32
DMASS	DTS Source Address Register	DTSA067	XXXX XXXX$_H$	32	FFFF A0C0$_H$	0	32
DMASS	DTS Destination Address Register	DTDA067	XXXX XXXX$_H$	32	FFFF A0C4$_H$	0	32
DMASS	DTS Transfer Count Register	DTTC067	XXXX XXXX$_H$	32	FFFF A0C8$_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT067	XXXX XXXX$_H$	32	FFFF A0CC$_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA067	XXXX XXXX$_H$	32	FFFF A0D0$_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA067	XXXX XXXX$_H$	32	FFFF A0D4$_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC067	XXXX XXXX$_H$	32	FFFF A0D8$_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC067	XXXX XXXX$_H$	32	FFFF A0DC$_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL067	0000 0000$_H$	32	FFFF A0E0$_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST067	0000 0000$_H$	32	FFFF A0E4$_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS067	0000 0000$_H$	32	FFFF A0E8$_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC067	0000 0000$_H$	32	FFFF A0EC$_H$	0	32
DMASS	DTS Source Address Register	DTSA068	XXXX XXXX$_H$	32	FFFF A100$_H$	0	32
DMASS	DTS Destination Address Register	DTDA068	XXXX XXXX$_H$	32	FFFF A104$_H$	0	32
DMASS	DTS Transfer Count Register	DTTC068	XXXX XXXX$_H$	32	FFFF A108$_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT068	XXXX XXXX$_H$	32	FFFF A10C$_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA068	XXXX XXXX$_H$	32	FFFF A110$_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA068	XXXX XXXX$_H$	32	FFFF A114$_H$	0	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
DMASS	DTS Reload Transfer Count Register	DTRTC068	XXXX $XXXX_H$	32	FFFF $A118_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC068	XXXX $XXXX_H$	32	FFFF $A11C_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL068	0000 0000_H	32	FFFF $A120_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST068	0000 0000_H	32	FFFF $A124_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS068	0000 0000_H	32	FFFF $A128_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC068	0000 0000_H	32	FFFF $A12C_H$	0	32
DMASS	DTS Source Address Register	DTSA069	XXXX $XXXX_H$	32	FFFF $A140_H$	0	32
DMASS	DTS Destination Address Register	DTDA069	XXXX $XXXX_H$	32	FFFF $A144_H$	0	32
DMASS	DTS Transfer Count Register	DTTC069	XXXX $XXXX_H$	32	FFFF $A148_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT069	XXXX $XXXX_H$	32	FFFF $A14C_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA069	XXXX $XXXX_H$	32	FFFF $A150_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA069	XXXX $XXXX_H$	32	FFFF $A154_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC069	XXXX $XXXX_H$	32	FFFF $A158_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC069	XXXX $XXXX_H$	32	FFFF $A15C_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL069	0000 0000_H	32	FFFF $A160_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST069	0000 0000_H	32	FFFF $A164_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS069	0000 0000_H	32	FFFF $A168_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC069	0000 0000_H	32	FFFF $A16C_H$	0	32
DMASS	DTS Source Address Register	DTSA070	XXXX $XXXX_H$	32	FFFF $A180_H$	0	32
DMASS	DTS Destination Address Register	DTDA070	XXXX $XXXX_H$	32	FFFF $A184_H$	0	32
DMASS	DTS Transfer Count Register	DTTC070	XXXX $XXXX_H$	32	FFFF $A188_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT070	XXXX $XXXX_H$	32	FFFF $A18C_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA070	XXXX $XXXX_H$	32	FFFF $A190_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA070	XXXX $XXXX_H$	32	FFFF $A194_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC070	XXXX $XXXX_H$	32	FFFF $A198_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC070	XXXX $XXXX_H$	32	FFFF $A19C_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL070	0000 0000_H	32	FFFF $A1A0_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST070	0000 0000_H	32	FFFF $A1A4_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS070	0000 0000_H	32	FFFF $A1A8_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC070	0000 0000_H	32	FFFF $A1AC_H$	0	32
DMASS	DTS Source Address Register	DTSA071	XXXX $XXXX_H$	32	FFFF $A1C0_H$	0	32
DMASS	DTS Destination Address Register	DTDA071	XXXX $XXXX_H$	32	FFFF $A1C4_H$	0	32
DMASS	DTS Transfer Count Register	DTTC071	XXXX $XXXX_H$	32	FFFF $A1C8_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT071	XXXX $XXXX_H$	32	FFFF $A1CC_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA071	XXXX $XXXX_H$	32	FFFF $A1D0_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA071	XXXX $XXXX_H$	32	FFFF $A1D4_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC071	XXXX $XXXX_H$	32	FFFF $A1D8_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC071	XXXX $XXXX_H$	32	FFFF $A1DC_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL071	0000 0000_H	32	FFFF $A1E0_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST071	0000 0000_H	32	FFFF $A1E4_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS071	0000 0000_H	32	FFFF $A1E8_H$	0	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
DMASS	DTSFSL Transfer Request Clear Register	DTFSC071	0000 0000_H	32	FFFF $A1EC_H$	0	32
DMASS	DTS Source Address Register	DTSA072	XXXX $XXXX_H$	32	FFFF $A200_H$	0	32
DMASS	DTS Destination Address Register	DTDA072	XXXX $XXXX_H$	32	FFFF $A204_H$	0	32
DMASS	DTS Transfer Count Register	DTTC072	XXXX $XXXX_H$	32	FFFF $A208_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT072	XXXX $XXXX_H$	32	FFFF $A20C_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA072	XXXX $XXXX_H$	32	FFFF $A210_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA072	XXXX $XXXX_H$	32	FFFF $A214_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC072	XXXX $XXXX_H$	32	FFFF $A218_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC072	XXXX $XXXX_H$	32	FFFF $A21C_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL072	0000 0000_H	32	FFFF $A220_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST072	0000 0000_H	32	FFFF $A224_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS072	0000 0000_H	32	FFFF $A228_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC072	0000 0000_H	32	FFFF $A22C_H$	0	32
DMASS	DTS Source Address Register	DTSA073	XXXX $XXXX_H$	32	FFFF $A240_H$	0	32
DMASS	DTS Destination Address Register	DTDA073	XXXX $XXXX_H$	32	FFFF $A244_H$	0	32
DMASS	DTS Transfer Count Register	DTTC073	XXXX $XXXX_H$	32	FFFF $A248_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT073	XXXX $XXXX_H$	32	FFFF $A24C_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA073	XXXX $XXXX_H$	32	FFFF $A250_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA073	XXXX $XXXX_H$	32	FFFF $A254_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC073	XXXX $XXXX_H$	32	FFFF $A258_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC073	XXXX $XXXX_H$	32	FFFF $A25C_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL073	0000 0000_H	32	FFFF $A260_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST073	0000 0000_H	32	FFFF $A264_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS073	0000 0000_H	32	FFFF $A268_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC073	0000 0000_H	32	FFFF $A26C_H$	0	32
DMASS	DTS Source Address Register	DTSA074	XXXX $XXXX_H$	32	FFFF $A280_H$	0	32
DMASS	DTS Destination Address Register	DTDA074	XXXX $XXXX_H$	32	FFFF $A284_H$	0	32
DMASS	DTS Transfer Count Register	DTTC074	XXXX $XXXX_H$	32	FFFF $A288_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT074	XXXX $XXXX_H$	32	FFFF $A28C_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA074	XXXX $XXXX_H$	32	FFFF $A290_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA074	XXXX $XXXX_H$	32	FFFF $A294_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC074	XXXX $XXXX_H$	32	FFFF $A298_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC074	XXXX $XXXX_H$	32	FFFF $A29C_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL074	0000 0000_H	32	FFFF $A2A0_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST074	0000 0000_H	32	FFFF $A2A4_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS074	0000 0000_H	32	FFFF $A2A8_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC074	0000 0000_H	32	FFFF $A2AC_H$	0	32
DMASS	DTS Source Address Register	DTSA075	XXXX $XXXX_H$	32	FFFF $A2C0_H$	0	32
DMASS	DTS Destination Address Register	DTDA075	XXXX $XXXX_H$	32	FFFF $A2C4_H$	0	32
DMASS	DTS Transfer Count Register	DTTC075	XXXX $XXXX_H$	32	FFFF $A2C8_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT075	XXXX $XXXX_H$	32	FFFF $A2CC_H$	0	32

(167/201)

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
DMASS	DTS Reload Source Address Register	DTRSA075	XXXX XXXX$_H$	32	FFFF A2D0$_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA075	XXXX XXXX$_H$	32	FFFF A2D4$_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC075	XXXX XXXX$_H$	32	FFFF A2D8$_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC075	XXXX XXXX$_H$	32	FFFF A2DC$_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL075	0000 0000$_H$	32	FFFF A2E0$_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST075	0000 0000$_H$	32	FFFF A2E4$_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS075	0000 0000$_H$	32	FFFF A2E8$_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC075	0000 0000$_H$	32	FFFF A2EC$_H$	0	32
DMASS	DTS Source Address Register	DTSA076	XXXX XXXX$_H$	32	FFFF A300$_H$	0	32
DMASS	DTS Destination Address Register	DTDA076	XXXX XXXX$_H$	32	FFFF A304$_H$	0	32
DMASS	DTS Transfer Count Register	DTTC076	XXXX XXXX$_H$	32	FFFF A308$_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT076	XXXX XXXX$_H$	32	FFFF A30C$_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA076	XXXX XXXX$_H$	32	FFFF A310$_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA076	XXXX XXXX$_H$	32	FFFF A314$_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC076	XXXX XXXX$_H$	32	FFFF A318$_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC076	XXXX XXXX$_H$	32	FFFF A31C$_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL076	0000 0000$_H$	32	FFFF A320$_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST076	0000 0000$_H$	32	FFFF A324$_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS076	0000 0000$_H$	32	FFFF A328$_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC076	0000 0000$_H$	32	FFFF A32C$_H$	0	32
DMASS	DTS Source Address Register	DTSA077	XXXX XXXX$_H$	32	FFFF A340$_H$	0	32
DMASS	DTS Destination Address Register	DTDA077	XXXX XXXX$_H$	32	FFFF A344$_H$	0	32
DMASS	DTS Transfer Count Register	DTTC077	XXXX XXXX$_H$	32	FFFF A348$_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT077	XXXX XXXX$_H$	32	FFFF A34C$_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA077	XXXX XXXX$_H$	32	FFFF A350$_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA077	XXXX XXXX$_H$	32	FFFF A354$_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC077	XXXX XXXX$_H$	32	FFFF A358$_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC077	XXXX XXXX$_H$	32	FFFF A35C$_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL077	0000 0000$_H$	32	FFFF A360$_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST077	0000 0000$_H$	32	FFFF A364$_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS077	0000 0000$_H$	32	FFFF A368$_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC077	0000 0000$_H$	32	FFFF A36C$_H$	0	32
DMASS	DTS Source Address Register	DTSA078	XXXX XXXX$_H$	32	FFFF A380$_H$	0	32
DMASS	DTS Destination Address Register	DTDA078	XXXX XXXX$_H$	32	FFFF A384$_H$	0	32
DMASS	DTS Transfer Count Register	DTTC078	XXXX XXXX$_H$	32	FFFF A388$_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT078	XXXX XXXX$_H$	32	FFFF A38C$_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA078	XXXX XXXX$_H$	32	FFFF A390$_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA078	XXXX XXXX$_H$	32	FFFF A394$_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC078	XXXX XXXX$_H$	32	FFFF A398$_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC078	XXXX XXXX$_H$	32	FFFF A39C$_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL078	0000 0000$_H$	32	FFFF A3A0$_H$	0	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
DMASS	DTSFSL Transfer Request Status Register	DTFST078	0000 0000$_H$	32	FFFF A3A4$_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS078	0000 0000$_H$	32	FFFF A3A8$_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC078	0000 0000$_H$	32	FFFF A3AC$_H$	0	32
DMASS	DTS Source Address Register	DTSA079	XXXX XXXX$_H$	32	FFFF A3C0$_H$	0	32
DMASS	DTS Destination Address Register	DTDA079	XXXX XXXX$_H$	32	FFFF A3C4$_H$	0	32
DMASS	DTS Transfer Count Register	DTTC079	XXXX XXXX$_H$	32	FFFF A3C8$_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT079	XXXX XXXX$_H$	32	FFFF A3CC$_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA079	XXXX XXXX$_H$	32	FFFF A3D0$_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA079	XXXX XXXX$_H$	32	FFFF A3D4$_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC079	XXXX XXXX$_H$	32	FFFF A3D8$_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC079	XXXX XXXX$_H$	32	FFFF A3DC$_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL079	0000 0000$_H$	32	FFFF A3E0$_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST079	0000 0000$_H$	32	FFFF A3E4$_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS079	0000 0000$_H$	32	FFFF A3E8$_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC079	0000 0000$_H$	32	FFFF A3EC$_H$	0	32
DMASS	DTS Source Address Register	DTSA080	XXXX XXXX$_H$	32	FFFF A400$_H$	0	32
DMASS	DTS Destination Address Register	DTDA080	XXXX XXXX$_H$	32	FFFF A404$_H$	0	32
DMASS	DTS Transfer Count Register	DTTC080	XXXX XXXX$_H$	32	FFFF A408$_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT080	XXXX XXXX$_H$	32	FFFF A40C$_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA080	XXXX XXXX$_H$	32	FFFF A410$_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA080	XXXX XXXX$_H$	32	FFFF A414$_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC080	XXXX XXXX$_H$	32	FFFF A418$_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC080	XXXX XXXX$_H$	32	FFFF A41C$_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL080	0000 0000$_H$	32	FFFF A420$_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST080	0000 0000$_H$	32	FFFF A424$_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS080	0000 0000$_H$	32	FFFF A428$_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC080	0000 0000$_H$	32	FFFF A42C$_H$	0	32
DMASS	DTS Source Address Register	DTSA081	XXXX XXXX$_H$	32	FFFF A440$_H$	0	32
DMASS	DTS Destination Address Register	DTDA081	XXXX XXXX$_H$	32	FFFF A444$_H$	0	32
DMASS	DTS Transfer Count Register	DTTC081	XXXX XXXX$_H$	32	FFFF A448$_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT081	XXXX XXXX$_H$	32	FFFF A44C$_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA081	XXXX XXXX$_H$	32	FFFF A450$_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA081	XXXX XXXX$_H$	32	FFFF A454$_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC081	XXXX XXXX$_H$	32	FFFF A458$_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC081	XXXX XXXX$_H$	32	FFFF A45C$_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL081	0000 0000$_H$	32	FFFF A460$_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST081	0000 0000$_H$	32	FFFF A464$_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS081	0000 0000$_H$	32	FFFF A468$_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC081	0000 0000$_H$	32	FFFF A46C$_H$	0	32
DMASS	DTS Source Address Register	DTSA082	XXXX XXXX$_H$	32	FFFF A480$_H$	0	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
DMASS	DTS Destination Address Register	DTDA082	XXXX XXXX$_H$	32	FFFF A484$_H$	0	32
DMASS	DTS Transfer Count Register	DTTC082	XXXX XXXX$_H$	32	FFFF A488$_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT082	XXXX XXXX$_H$	32	FFFF A48C$_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA082	XXXX XXXX$_H$	32	FFFF A490$_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA082	XXXX XXXX$_H$	32	FFFF A494$_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC082	XXXX XXXX$_H$	32	FFFF A498$_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC082	XXXX XXXX$_H$	32	FFFF A49C$_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL082	0000 0000$_H$	32	FFFF A4A0$_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST082	0000 0000$_H$	32	FFFF A4A4$_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS082	0000 0000$_H$	32	FFFF A4A8$_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC082	0000 0000$_H$	32	FFFF A4AC$_H$	0	32
DMASS	DTS Source Address Register	DTSA083	XXXX XXXX$_H$	32	FFFF A4C0$_H$	0	32
DMASS	DTS Destination Address Register	DTDA083	XXXX XXXX$_H$	32	FFFF A4C4$_H$	0	32
DMASS	DTS Transfer Count Register	DTTC083	XXXX XXXX$_H$	32	FFFF A4C8$_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT083	XXXX XXXX$_H$	32	FFFF A4CC$_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA083	XXXX XXXX$_H$	32	FFFF A4D0$_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA083	XXXX XXXX$_H$	32	FFFF A4D4$_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC083	XXXX XXXX$_H$	32	FFFF A4D8$_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC083	XXXX XXXX$_H$	32	FFFF A4DC$_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL083	0000 0000$_H$	32	FFFF A4E0$_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST083	0000 0000$_H$	32	FFFF A4E4$_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS083	0000 0000$_H$	32	FFFF A4E8$_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC083	0000 0000$_H$	32	FFFF A4EC$_H$	0	32
DMASS	DTS Source Address Register	DTSA084	XXXX XXXX$_H$	32	FFFF A500$_H$	0	32
DMASS	DTS Destination Address Register	DTDA084	XXXX XXXX$_H$	32	FFFF A504$_H$	0	32
DMASS	DTS Transfer Count Register	DTTC084	XXXX XXXX$_H$	32	FFFF A508$_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT084	XXXX XXXX$_H$	32	FFFF A50C$_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA084	XXXX XXXX$_H$	32	FFFF A510$_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA084	XXXX XXXX$_H$	32	FFFF A514$_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC084	XXXX XXXX$_H$	32	FFFF A518$_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC084	XXXX XXXX$_H$	32	FFFF A51C$_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL084	0000 0000$_H$	32	FFFF A520$_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST084	0000 0000$_H$	32	FFFF A524$_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS084	0000 0000$_H$	32	FFFF A528$_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC084	0000 0000$_H$	32	FFFF A52C$_H$	0	32
DMASS	DTS Source Address Register	DTSA085	XXXX XXXX$_H$	32	FFFF A540$_H$	0	32
DMASS	DTS Destination Address Register	DTDA085	XXXX XXXX$_H$	32	FFFF A544$_H$	0	32
DMASS	DTS Transfer Count Register	DTTC085	XXXX XXXX$_H$	32	FFFF A548$_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT085	XXXX XXXX$_H$	32	FFFF A54C$_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA085	XXXX XXXX$_H$	32	FFFF A550$_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA085	XXXX XXXX$_H$	32	FFFF A554$_H$	0	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
DMASS	DTS Reload Transfer Count Register	DTRTC085	XXXX $XXXX_H$	32	FFFF $A558_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC085	XXXX $XXXX_H$	32	FFFF $A55C_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL085	0000 0000_H	32	FFFF $A560_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST085	0000 0000_H	32	FFFF $A564_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS085	0000 0000_H	32	FFFF $A568_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC085	0000 0000_H	32	FFFF $A56C_H$	0	32
DMASS	DTS Source Address Register	DTSA086	XXXX $XXXX_H$	32	FFFF $A580_H$	0	32
DMASS	DTS Destination Address Register	DTDA086	XXXX $XXXX_H$	32	FFFF $A584_H$	0	32
DMASS	DTS Transfer Count Register	DTTC086	XXXX $XXXX_H$	32	FFFF $A588_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT086	XXXX $XXXX_H$	32	FFFF $A58C_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA086	XXXX $XXXX_H$	32	FFFF $A590_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA086	XXXX $XXXX_H$	32	FFFF $A594_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC086	XXXX $XXXX_H$	32	FFFF $A598_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC086	XXXX $XXXX_H$	32	FFFF $A59C_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL086	0000 0000_H	32	FFFF $A5A0_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST086	0000 0000_H	32	FFFF $A5A4_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS086	0000 0000_H	32	FFFF $A5A8_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC086	0000 0000_H	32	FFFF $A5AC_H$	0	32
DMASS	DTS Source Address Register	DTSA087	XXXX $XXXX_H$	32	FFFF $A5C0_H$	0	32
DMASS	DTS Destination Address Register	DTDA087	XXXX $XXXX_H$	32	FFFF $A5C4_H$	0	32
DMASS	DTS Transfer Count Register	DTTC087	XXXX $XXXX_H$	32	FFFF $A5C8_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT087	XXXX $XXXX_H$	32	FFFF $A5CC_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA087	XXXX $XXXX_H$	32	FFFF $A5D0_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA087	XXXX $XXXX_H$	32	FFFF $A5D4_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC087	XXXX $XXXX_H$	32	FFFF $A5D8_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC087	XXXX $XXXX_H$	32	FFFF $A5DC_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL087	0000 0000_H	32	FFFF $A5E0_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST087	0000 0000_H	32	FFFF $A5E4_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS087	0000 0000_H	32	FFFF $A5E8_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC087	0000 0000_H	32	FFFF $A5EC_H$	0	32
DMASS	DTS Source Address Register	DTSA088	XXXX $XXXX_H$	32	FFFF $A600_H$	0	32
DMASS	DTS Destination Address Register	DTDA088	XXXX $XXXX_H$	32	FFFF $A604_H$	0	32
DMASS	DTS Transfer Count Register	DTTC088	XXXX $XXXX_H$	32	FFFF $A608_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT088	XXXX $XXXX_H$	32	FFFF $A60C_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA088	XXXX $XXXX_H$	32	FFFF $A610_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA088	XXXX $XXXX_H$	32	FFFF $A614_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC088	XXXX $XXXX_H$	32	FFFF $A618_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC088	XXXX $XXXX_H$	32	FFFF $A61C_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL088	0000 0000_H	32	FFFF $A620_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST088	0000 0000_H	32	FFFF $A624_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS088	0000 0000_H	32	FFFF $A628_H$	0	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
DMASS	DTSFSL Transfer Request Clear Register	DTFSC088	0000 0000_H	32	FFFF $A62C_H$	0	32
DMASS	DTS Source Address Register	DTSA089	XXXX $XXXX_H$	32	FFFF $A640_H$	0	32
DMASS	DTS Destination Address Register	DTDA089	XXXX $XXXX_H$	32	FFFF $A644_H$	0	32
DMASS	DTS Transfer Count Register	DTTC089	XXXX $XXXX_H$	32	FFFF $A648_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT089	XXXX $XXXX_H$	32	FFFF $A64C_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA089	XXXX $XXXX_H$	32	FFFF $A650_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA089	XXXX $XXXX_H$	32	FFFF $A654_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC089	XXXX $XXXX_H$	32	FFFF $A658_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC089	XXXX $XXXX_H$	32	FFFF $A65C_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL089	0000 0000_H	32	FFFF $A660_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST089	0000 0000_H	32	FFFF $A664_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS089	0000 0000_H	32	FFFF $A668_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC089	0000 0000_H	32	FFFF $A66C_H$	0	32
DMASS	DTS Source Address Register	DTSA090	XXXX $XXXX_H$	32	FFFF $A680_H$	0	32
DMASS	DTS Destination Address Register	DTDA090	XXXX $XXXX_H$	32	FFFF $A684_H$	0	32
DMASS	DTS Transfer Count Register	DTTC090	XXXX $XXXX_H$	32	FFFF $A688_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT090	XXXX $XXXX_H$	32	FFFF $A68C_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA090	XXXX $XXXX_H$	32	FFFF $A690_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA090	XXXX $XXXX_H$	32	FFFF $A694_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC090	XXXX $XXXX_H$	32	FFFF $A698_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC090	XXXX $XXXX_H$	32	FFFF $A69C_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL090	0000 0000_H	32	FFFF $A6A0_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST090	0000 0000_H	32	FFFF $A6A4_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS090	0000 0000_H	32	FFFF $A6A8_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC090	0000 0000_H	32	FFFF $A6AC_H$	0	32
DMASS	DTS Source Address Register	DTSA091	XXXX $XXXX_H$	32	FFFF $A6C0_H$	0	32
DMASS	DTS Destination Address Register	DTDA091	XXXX $XXXX_H$	32	FFFF $A6C4_H$	0	32
DMASS	DTS Transfer Count Register	DTTC091	XXXX $XXXX_H$	32	FFFF $A6C8_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT091	XXXX $XXXX_H$	32	FFFF $A6CC_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA091	XXXX $XXXX_H$	32	FFFF $A6D0_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA091	XXXX $XXXX_H$	32	FFFF $A6D4_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC091	XXXX $XXXX_H$	32	FFFF $A6D8_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC091	XXXX $XXXX_H$	32	FFFF $A6DC_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL091	0000 0000_H	32	FFFF $A6E0_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST091	0000 0000_H	32	FFFF $A6E4_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS091	0000 0000_H	32	FFFF $A6E8_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC091	0000 0000_H	32	FFFF $A6EC_H$	0	32
DMASS	DTS Source Address Register	DTSA092	XXXX $XXXX_H$	32	FFFF $A700_H$	0	32
DMASS	DTS Destination Address Register	DTDA092	XXXX $XXXX_H$	32	FFFF $A704_H$	0	32
DMASS	DTS Transfer Count Register	DTTC092	XXXX $XXXX_H$	32	FFFF $A708_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT092	XXXX $XXXX_H$	32	FFFF $A70C_H$	0	32

(172/201)

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
DMASS	DTS Reload Source Address Register	DTRSA092	XXXX XXXX$_H$	32	FFFF A710$_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA092	XXXX XXXX$_H$	32	FFFF A714$_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC092	XXXX XXXX$_H$	32	FFFF A718$_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC092	XXXX XXXX$_H$	32	FFFF A71C$_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL092	0000 0000$_H$	32	FFFF A720$_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST092	0000 0000$_H$	32	FFFF A724$_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS092	0000 0000$_H$	32	FFFF A728$_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC092	0000 0000$_H$	32	FFFF A72C$_H$	0	32
DMASS	DTS Source Address Register	DTSA093	XXXX XXXX$_H$	32	FFFF A740$_H$	0	32
DMASS	DTS Destination Address Register	DTDA093	XXXX XXXX$_H$	32	FFFF A744$_H$	0	32
DMASS	DTS Transfer Count Register	DTTC093	XXXX XXXX$_H$	32	FFFF A748$_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT093	XXXX XXXX$_H$	32	FFFF A74C$_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA093	XXXX XXXX$_H$	32	FFFF A750$_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA093	XXXX XXXX$_H$	32	FFFF A754$_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC093	XXXX XXXX$_H$	32	FFFF A758$_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC093	XXXX XXXX$_H$	32	FFFF A75C$_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL093	0000 0000$_H$	32	FFFF A760$_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST093	0000 0000$_H$	32	FFFF A764$_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS093	0000 0000$_H$	32	FFFF A768$_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC093	0000 0000$_H$	32	FFFF A76C$_H$	0	32
DMASS	DTS Source Address Register	DTSA094	XXXX XXXX$_H$	32	FFFF A780$_H$	0	32
DMASS	DTS Destination Address Register	DTDA094	XXXX XXXX$_H$	32	FFFF A784$_H$	0	32
DMASS	DTS Transfer Count Register	DTTC094	XXXX XXXX$_H$	32	FFFF A788$_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT094	XXXX XXXX$_H$	32	FFFF A78C$_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA094	XXXX XXXX$_H$	32	FFFF A790$_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA094	XXXX XXXX$_H$	32	FFFF A794$_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC094	XXXX XXXX$_H$	32	FFFF A798$_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC094	XXXX XXXX$_H$	32	FFFF A79C$_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL094	0000 0000$_H$	32	FFFF A7A0$_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST094	0000 0000$_H$	32	FFFF A7A4$_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS094	0000 0000$_H$	32	FFFF A7A8$_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC094	0000 0000$_H$	32	FFFF A7AC$_H$	0	32
DMASS	DTS Source Address Register	DTSA095	XXXX XXXX$_H$	32	FFFF A7C0$_H$	0	32
DMASS	DTS Destination Address Register	DTDA095	XXXX XXXX$_H$	32	FFFF A7C4$_H$	0	32
DMASS	DTS Transfer Count Register	DTTC095	XXXX XXXX$_H$	32	FFFF A7C8$_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT095	XXXX XXXX$_H$	32	FFFF A7CC$_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA095	XXXX XXXX$_H$	32	FFFF A7D0$_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA095	XXXX XXXX$_H$	32	FFFF A7D4$_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC095	XXXX XXXX$_H$	32	FFFF A7D8$_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC095	XXXX XXXX$_H$	32	FFFF A7DC$_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL095	0000 0000$_H$	32	FFFF A7E0$_H$	0	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
DMASS	DTSFSL Transfer Request Status Register	DTFST095	0000 0000_H	32	FFFF $A7E4_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS095	0000 0000_H	32	FFFF $A7E8_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC095	0000 0000_H	32	FFFF $A7EC_H$	0	32
DMASS	DTS Source Address Register	DTSA096	XXXX $XXXX_H$	32	FFFF $A800_H$	0	32
DMASS	DTS Destination Address Register	DTDA096	XXXX $XXXX_H$	32	FFFF $A804_H$	0	32
DMASS	DTS Transfer Count Register	DTTC096	XXXX $XXXX_H$	32	FFFF $A808_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT096	XXXX $XXXX_H$	32	FFFF $A80C_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA096	XXXX $XXXX_H$	32	FFFF $A810_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA096	XXXX $XXXX_H$	32	FFFF $A814_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC096	XXXX $XXXX_H$	32	FFFF $A818_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC096	XXXX $XXXX_H$	32	FFFF $A81C_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL096	0000 0000_H	32	FFFF $A820_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST096	0000 0000_H	32	FFFF $A824_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS096	0000 0000_H	32	FFFF $A828_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC096	0000 0000_H	32	FFFF $A82C_H$	0	32
DMASS	DTS Source Address Register	DTSA097	XXXX $XXXX_H$	32	FFFF $A840_H$	0	32
DMASS	DTS Destination Address Register	DTDA097	XXXX $XXXX_H$	32	FFFF $A844_H$	0	32
DMASS	DTS Transfer Count Register	DTTC097	XXXX $XXXX_H$	32	FFFF $A848_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT097	XXXX $XXXX_H$	32	FFFF $A84C_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA097	XXXX $XXXX_H$	32	FFFF $A850_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA097	XXXX $XXXX_H$	32	FFFF $A854_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC097	XXXX $XXXX_H$	32	FFFF $A858_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC097	XXXX $XXXX_H$	32	FFFF $A85C_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL097	0000 0000_H	32	FFFF $A860_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST097	0000 0000_H	32	FFFF $A864_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS097	0000 0000_H	32	FFFF $A868_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC097	0000 0000_H	32	FFFF $A86C_H$	0	32
DMASS	DTS ???????????	DTSA098	XXXX $XXXX_H$	32	FFFF $A880_H$	0	32
DMASS	DTS Source Address Register	DTDA098	XXXX $XXXX_H$	32	FFFF $A884_H$	0	32
DMASS	DTS Destination Address Register	DTTC098	XXXX $XXXX_H$	32	FFFF $A888_H$	0	32
DMASS	DTS Transfer Count Register	DTTCT098	XXXX $XXXX_H$	32	FFFF $A88C_H$	0	32
DMASS	DTS Transfer Control Register	DTRSA098	XXXX $XXXX_H$	32	FFFF $A890_H$	0	32
DMASS	DTS Reload Source Address Register	DTRDA098	XXXX $XXXX_H$	32	FFFF $A894_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRTC098	XXXX $XXXX_H$	32	FFFF $A898_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTTCC098	XXXX $XXXX_H$	32	FFFF $A89C_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTFSL098	0000 0000_H	32	FFFF $A8A0_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFST098	0000 0000_H	32	FFFF $A8A4_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFSS098	0000 0000_H	32	FFFF $A8A8_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSC098	0000 0000_H	32	FFFF $A8AC_H$	0	32
DMASS	DTS Source Address Register	DTSA099	XXXX $XXXX_H$	32	FFFF $A8C0_H$	0	32
DMASS	DTS Destination Address Register	DTDA099	XXXX $XXXX_H$	32	FFFF $A8C4_H$	0	32

(174/201)

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
DMASS	DTS Transfer Count Register	DTTC099	XXXX $XXXX_H$	32	FFFF $A8C8_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT099	XXXX $XXXX_H$	32	FFFF $A8CC_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA099	XXXX $XXXX_H$	32	FFFF $A8D0_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA099	XXXX $XXXX_H$	32	FFFF $A8D4_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC099	XXXX $XXXX_H$	32	FFFF $A8D8_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC099	XXXX $XXXX_H$	32	FFFF $A8DC_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL099	0000 0000_H	32	FFFF $A8E0_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST099	0000 0000_H	32	FFFF $A8E4_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS099	0000 0000_H	32	FFFF $A8E8_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC099	0000 0000_H	32	FFFF $A8EC_H$	0	32
DMASS	DTS Source Address Register	DTSA100	XXXX $XXXX_H$	32	FFFF $A900_H$	0	32
DMASS	DTS Destination Address Register	DTDA100	XXXX $XXXX_H$	32	FFFF $A904_H$	0	32
DMASS	DTS Transfer Count Register	DTTC100	XXXX $XXXX_H$	32	FFFF $A908_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT100	XXXX $XXXX_H$	32	FFFF $A90C_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA100	XXXX $XXXX_H$	32	FFFF $A910_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA100	XXXX $XXXX_H$	32	FFFF $A914_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC100	XXXX $XXXX_H$	32	FFFF $A918_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC100	XXXX $XXXX_H$	32	FFFF $A91C_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL100	0000 0000_H	32	FFFF $A920_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST100	0000 0000_H	32	FFFF $A924_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS100	0000 0000_H	32	FFFF $A928_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC100	0000 0000_H	32	FFFF $A92C_H$	0	32
DMASS	DTS Source Address Register	DTSA101	XXXX $XXXX_H$	32	FFFF $A940_H$	0	32
DMASS	DTS Destination Address Register	DTDA101	XXXX $XXXX_H$	32	FFFF $A944_H$	0	32
DMASS	DTS Transfer Count Register	DTTC101	XXXX $XXXX_H$	32	FFFF $A948_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT101	XXXX $XXXX_H$	32	FFFF $A94C_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA101	XXXX $XXXX_H$	32	FFFF $A950_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA101	XXXX $XXXX_H$	32	FFFF $A954_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC101	XXXX $XXXX_H$	32	FFFF $A958_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC101	XXXX $XXXX_H$	32	FFFF $A95C_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL101	0000 0000_H	32	FFFF $A960_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST101	0000 0000_H	32	FFFF $A964_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS101	0000 0000_H	32	FFFF $A968_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC101	0000 0000_H	32	FFFF $A96C_H$	0	32
DMASS	DTS Source Address Register	DTSA102	XXXX $XXXX_H$	32	FFFF $A980_H$	0	32
DMASS	DTS Destination Address Register	DTDA102	XXXX $XXXX_H$	32	FFFF $A984_H$	0	32
DMASS	DTS Transfer Count Register	DTTC102	XXXX $XXXX_H$	32	FFFF $A988_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT102	XXXX $XXXX_H$	32	FFFF $A98C_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA102	XXXX $XXXX_H$	32	FFFF $A990_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA102	XXXX $XXXX_H$	32	FFFF $A994_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC102	XXXX $XXXX_H$	32	FFFF $A998_H$	0	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
DMASS	DTS Transfer Count Compare Register	DTTCC102	XXXX $XXXX_H$	32	FFFF $A99C_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL102	$0000\ 0000_H$	32	FFFF $A9A0_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST102	$0000\ 0000_H$	32	FFFF $A9A4_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS102	$0000\ 0000_H$	32	FFFF $A9A8_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC102	$0000\ 0000_H$	32	FFFF $A9AC_H$	0	32
DMASS	DTS Source Address Register	DTSA103	XXXX $XXXX_H$	32	FFFF $A9C0_H$	0	32
DMASS	DTS Destination Address Register	DTDA103	XXXX $XXXX_H$	32	FFFF $A9C4_H$	0	32
DMASS	DTS Transfer Count Register	DTTC103	XXXX $XXXX_H$	32	FFFF $A9C8_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT103	XXXX $XXXX_H$	32	FFFF $A9CC_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA103	XXXX $XXXX_H$	32	FFFF $A9D0_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA103	XXXX $XXXX_H$	32	FFFF $A9D4_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC103	XXXX $XXXX_H$	32	FFFF $A9D8_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC103	XXXX $XXXX_H$	32	FFFF $A9DC_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL103	$0000\ 0000_H$	32	FFFF $A9E0_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST103	$0000\ 0000_H$	32	FFFF $A9E4_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS103	$0000\ 0000_H$	32	FFFF $A9E8_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC103	$0000\ 0000_H$	32	FFFF $A9EC_H$	0	32
DMASS	DTS Source Address Register	DTSA104	XXXX $XXXX_H$	32	FFFF $AA00_H$	0	32
DMASS	DTS Destination Address Register	DTDA104	XXXX $XXXX_H$	32	FFFF $AA04_H$	0	32
DMASS	DTS Transfer Count Register	DTTC104	XXXX $XXXX_H$	32	FFFF $AA08_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT104	XXXX $XXXX_H$	32	FFFF $AA0C_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA104	XXXX $XXXX_H$	32	FFFF $AA10_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA104	XXXX $XXXX_H$	32	FFFF $AA14_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC104	XXXX $XXXX_H$	32	FFFF $AA18_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC104	XXXX $XXXX_H$	32	FFFF $AA1C_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL104	$0000\ 0000_H$	32	FFFF $AA20_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST104	$0000\ 0000_H$	32	FFFF $AA24_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS104	$0000\ 0000_H$	32	FFFF $AA28_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC104	$0000\ 0000_H$	32	FFFF $AA2C_H$	0	32
DMASS	DTS Source Address Register	DTSA105	XXXX $XXXX_H$	32	FFFF $AA40_H$	0	32
DMASS	DTS Destination Address Register	DTDA105	XXXX $XXXX_H$	32	FFFF $AA44_H$	0	32
DMASS	DTS Transfer Count Register	DTTC105	XXXX $XXXX_H$	32	FFFF $AA48_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT105	XXXX $XXXX_H$	32	FFFF $AA4C_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA105	XXXX $XXXX_H$	32	FFFF $AA50_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA105	XXXX $XXXX_H$	32	FFFF $AA54_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC105	XXXX $XXXX_H$	32	FFFF $AA58_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC105	XXXX $XXXX_H$	32	FFFF $AA5C_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL105	$0000\ 0000_H$	32	FFFF $AA60_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST105	$0000\ 0000_H$	32	FFFF $AA64_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS105	$0000\ 0000_H$	32	FFFF $AA68_H$	0	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
DMASS	DTSFSL Transfer Request Clear Register	DTFSC105	0000 0000_H	32	FFFF $AA6C_H$	0	32
DMASS	DTS Source Address Register	DTSA106	XXXX $XXXX_H$	32	FFFF $AA80_H$	0	32
DMASS	DTS Destination Address Register	DTDA106	XXXX $XXXX_H$	32	FFFF $AA84_H$	0	32
DMASS	DTS Transfer Count Register	DTTC106	XXXX $XXXX_H$	32	FFFF $AA88_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT106	XXXX $XXXX_H$	32	FFFF $AA8C_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA106	XXXX $XXXX_H$	32	FFFF $AA90_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA106	XXXX $XXXX_H$	32	FFFF $AA94_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC106	XXXX $XXXX_H$	32	FFFF $AA98_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC106	XXXX $XXXX_H$	32	FFFF $AA9C_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL106	0000 0000_H	32	FFFF $AAA0_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST106	0000 0000_H	32	FFFF $AAA4_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS106	0000 0000_H	32	FFFF $AAA8_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC106	0000 0000_H	32	FFFF $AAAC_H$	0	32
DMASS	DTS Source Address Register	DTSA107	XXXX $XXXX_H$	32	FFFF $AAC0_H$	0	32
DMASS	DTS Destination Address Register	DTDA107	XXXX $XXXX_H$	32	FFFF $AAC4_H$	0	32
DMASS	DTS Transfer Count Register	DTTC107	XXXX $XXXX_H$	32	FFFF $AAC8_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT107	XXXX $XXXX_H$	32	FFFF $AACC_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA107	XXXX $XXXX_H$	32	FFFF $AAD0_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA107	XXXX $XXXX_H$	32	FFFF $AAD4_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC107	XXXX $XXXX_H$	32	FFFF $AAD8_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC107	XXXX $XXXX_H$	32	FFFF $AADC_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL107	0000 0000_H	32	FFFF $AAE0_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST107	0000 0000_H	32	FFFF $AAE4_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS107	0000 0000_H	32	FFFF $AAE8_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC107	0000 0000_H	32	FFFF $AAEC_H$	0	32
DMASS	DTS Source Address Register	DTSA108	XXXX $XXXX_H$	32	FFFF $AB00_H$	0	32
DMASS	DTS Destination Address Register	DTDA108	XXXX $XXXX_H$	32	FFFF $AB04_H$	0	32
DMASS	DTS Transfer Count Register	DTTC108	XXXX $XXXX_H$	32	FFFF $AB08_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT108	XXXX $XXXX_H$	32	FFFF $AB0C_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA108	XXXX $XXXX_H$	32	FFFF $AB10_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA108	XXXX $XXXX_H$	32	FFFF $AB14_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC108	XXXX $XXXX_H$	32	FFFF $AB18_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC108	XXXX $XXXX_H$	32	FFFF $AB1C_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL108	0000 0000_H	32	FFFF $AB20_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST108	0000 0000_H	32	FFFF $AB24_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS108	0000 0000_H	32	FFFF $AB28_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC108	0000 0000_H	32	FFFF $AB2C_H$	0	32
DMASS	DTS Source Address Register	DTSA109	XXXX $XXXX_H$	32	FFFF $AB40_H$	0	32
DMASS	DTS Destination Address Register	DTDA109	XXXX $XXXX_H$	32	FFFF $AB44_H$	0	32
DMASS	DTS Transfer Count Register	DTTC109	XXXX $XXXX_H$	32	FFFF $AB48_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT109	XXXX $XXXX_H$	32	FFFF $AB4C_H$	0	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
DMASS	DTS Reload Source Address Register	DTRSA109	XXXX XXXX$_H$	32	FFFF AB50$_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA109	XXXX XXXX$_H$	32	FFFF AB54$_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC109	XXXX XXXX$_H$	32	FFFF AB58$_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC109	XXXX XXXX$_H$	32	FFFF AB5C$_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL109	0000 0000$_H$	32	FFFF AB60$_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST109	0000 0000$_H$	32	FFFF AB64$_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS109	0000 0000$_H$	32	FFFF AB68$_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC109	0000 0000$_H$	32	FFFF AB6C$_H$	0	32
DMASS	DTS Source Address Register	DTSA110	XXXX XXXX$_H$	32	FFFF AB80$_H$	0	32
DMASS	DTS Destination Address Register	DTDA110	XXXX XXXX$_H$	32	FFFF AB84$_H$	0	32
DMASS	DTS Transfer Count Register	DTTC110	XXXX XXXX$_H$	32	FFFF AB88$_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT110	XXXX XXXX$_H$	32	FFFF AB8C$_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA110	XXXX XXXX$_H$	32	FFFF AB90$_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA110	XXXX XXXX$_H$	32	FFFF AB94$_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC110	XXXX XXXX$_H$	32	FFFF AB98$_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC110	XXXX XXXX$_H$	32	FFFF AB9C$_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL110	0000 0000$_H$	32	FFFF ABA0$_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST110	0000 0000$_H$	32	FFFF ABA4$_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS110	0000 0000$_H$	32	FFFF ABA8$_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC110	0000 0000$_H$	32	FFFF ABAC$_H$	0	32
DMASS	DTS Source Address Register	DTSA111	XXXX XXXX$_H$	32	FFFF ABC0$_H$	0	32
DMASS	DTS Destination Address Register	DTDA111	XXXX XXXX$_H$	32	FFFF ABC4$_H$	0	32
DMASS	DTS Transfer Count Register	DTTC111	XXXX XXXX$_H$	32	FFFF ABC8$_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT111	XXXX XXXX$_H$	32	FFFF ABCC$_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA111	XXXX XXXX$_H$	32	FFFF ABD0$_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA111	XXXX XXXX$_H$	32	FFFF ABD4$_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC111	XXXX XXXX$_H$	32	FFFF ABD8$_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC111	XXXX XXXX$_H$	32	FFFF ABDC$_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL111	0000 0000$_H$	32	FFFF ABE0$_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST111	0000 0000$_H$	32	FFFF ABE4$_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS111	0000 0000$_H$	32	FFFF ABE8$_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC111	0000 0000$_H$	32	FFFF ABEC$_H$	0	32
DMASS	DTS Source Address Register	DTSA112	XXXX XXXX$_H$	32	FFFF AC00$_H$	0	32
DMASS	DTS Destination Address Register	DTDA112	XXXX XXXX$_H$	32	FFFF AC04$_H$	0	32
DMASS	DTS Transfer Count Register	DTTC112	XXXX XXXX$_H$	32	FFFF AC08$_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT112	XXXX XXXX$_H$	32	FFFF AC0C$_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA112	XXXX XXXX$_H$	32	FFFF AC10$_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA112	XXXX XXXX$_H$	32	FFFF AC14$_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC112	XXXX XXXX$_H$	32	FFFF AC18$_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC112	XXXX XXXX$_H$	32	FFFF AC1C$_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL112	0000 0000$_H$	32	FFFF AC20$_H$	0	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
DMASS	DTSFSL Transfer Request Status Register	DTFST112	0000 0000_H	32	FFFF $AC24_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS112	0000 0000_H	32	FFFF $AC28_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC112	0000 0000_H	32	FFFF $AC2C_H$	0	32
DMASS	DTS Source Address Register	DTSA113	XXXX $XXXX_H$	32	FFFF $AC40_H$	0	32
DMASS	DTS Destination Address Register	DTDA113	XXXX $XXXX_H$	32	FFFF $AC44_H$	0	32
DMASS	DTS Transfer Count Register	DTTC113	XXXX $XXXX_H$	32	FFFF $AC48_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT113	XXXX $XXXX_H$	32	FFFF $AC4C_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA113	XXXX $XXXX_H$	32	FFFF $AC50_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA113	XXXX $XXXX_H$	32	FFFF $AC54_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC113	XXXX $XXXX_H$	32	FFFF $AC58_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC113	XXXX $XXXX_H$	32	FFFF $AC5C_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL113	0000 0000_H	32	FFFF $AC60_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST113	0000 0000_H	32	FFFF $AC64_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS113	0000 0000_H	32	FFFF $AC68_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC113	0000 0000_H	32	FFFF $AC6C_H$	0	32
DMASS	DTS Source Address Register	DTSA114	XXXX $XXXX_H$	32	FFFF $AC80_H$	0	32
DMASS	DTS Destination Address Register	DTDA114	XXXX $XXXX_H$	32	FFFF $AC84_H$	0	32
DMASS	DTS Transfer Count Register	DTTC114	XXXX $XXXX_H$	32	FFFF $AC88_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT114	XXXX $XXXX_H$	32	FFFF $AC8C_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA114	XXXX $XXXX_H$	32	FFFF $AC90_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA114	XXXX $XXXX_H$	32	FFFF $AC94_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC114	XXXX $XXXX_H$	32	FFFF $AC98_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC114	XXXX $XXXX_H$	32	FFFF $AC9C_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL114	0000 0000_H	32	FFFF $ACA0_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST114	0000 0000_H	32	FFFF $ACA4_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS114	0000 0000_H	32	FFFF $ACA8_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC114	0000 0000_H	32	FFFF $ACAC_H$	0	32
DMASS	DTS Source Address Register	DTSA115	XXXX $XXXX_H$	32	FFFF $ACC0_H$	0	32
DMASS	DTS Destination Address Register	DTDA115	XXXX $XXXX_H$	32	FFFF $ACC4_H$	0	32
DMASS	DTS Transfer Count Register	DTTC115	XXXX $XXXX_H$	32	FFFF $ACC8_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT115	XXXX $XXXX_H$	32	FFFF $ACCC_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA115	XXXX $XXXX_H$	32	FFFF $ACD0_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA115	XXXX $XXXX_H$	32	FFFF $ACD4_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC115	XXXX $XXXX_H$	32	FFFF $ACD8_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC115	XXXX $XXXX_H$	32	FFFF $ACDC_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL115	0000 0000_H	32	FFFF $ACE0_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST115	0000 0000_H	32	FFFF $ACE4_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS115	0000 0000_H	32	FFFF $ACE8_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC115	0000 0000_H	32	FFFF $ACEC_H$	0	32
DMASS	DTS Source Address Register	DTSA116	XXXX $XXXX_H$	32	FFFF $AD00_H$	0	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
DMASS	DTS Destination Address Register	DTDA116	XXXX XXXX$_H$	32	FFFF AD04$_H$	0	32
DMASS	DTS Transfer Count Register	DTTC116	XXXX XXXX$_H$	32	FFFF AD08$_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT116	XXXX XXXX$_H$	32	FFFF AD0C$_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA116	XXXX XXXX$_H$	32	FFFF AD10$_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA116	XXXX XXXX$_H$	32	FFFF AD14$_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC116	XXXX XXXX$_H$	32	FFFF AD18$_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC116	XXXX XXXX$_H$	32	FFFF AD1C$_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL116	0000 0000$_H$	32	FFFF AD20$_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST116	0000 0000$_H$	32	FFFF AD24$_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS116	0000 0000$_H$	32	FFFF AD28$_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC116	0000 0000$_H$	32	FFFF AD2C$_H$	0	32
DMASS	DTS Source Address Register	DTSA117	XXXX XXXX$_H$	32	FFFF AD40$_H$	0	32
DMASS	DTS Destination Address Register	DTDA117	XXXX XXXX$_H$	32	FFFF AD44$_H$	0	32
DMASS	DTS Transfer Count Register	DTTC117	XXXX XXXX$_H$	32	FFFF AD48$_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT117	XXXX XXXX$_H$	32	FFFF AD4C$_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA117	XXXX XXXX$_H$	32	FFFF AD50$_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA117	XXXX XXXX$_H$	32	FFFF AD54$_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC117	XXXX XXXX$_H$	32	FFFF AD58$_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC117	XXXX XXXX$_H$	32	FFFF AD5C$_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL117	0000 0000$_H$	32	FFFF AD60$_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST117	0000 0000$_H$	32	FFFF AD64$_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS117	0000 0000$_H$	32	FFFF AD68$_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC117	0000 0000$_H$	32	FFFF AD6C$_H$	0	32
DMASS	DTS Source Address Register	DTSA118	XXXX XXXX$_H$	32	FFFF AD80$_H$	0	32
DMASS	DTS Destination Address Register	DTDA118	XXXX XXXX$_H$	32	FFFF AD84$_H$	0	32
DMASS	DTS Transfer Count Register	DTTC118	XXXX XXXX$_H$	32	FFFF AD88$_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT118	XXXX XXXX$_H$	32	FFFF AD8C$_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA118	XXXX XXXX$_H$	32	FFFF AD90$_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA118	XXXX XXXX$_H$	32	FFFF AD94$_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC118	XXXX XXXX$_H$	32	FFFF AD98$_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC118	XXXX XXXX$_H$	32	FFFF AD9C$_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL118	0000 0000$_H$	32	FFFF ADA0$_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST118	0000 0000$_H$	32	FFFF ADA4$_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS118	0000 0000$_H$	32	FFFF ADA8$_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC118	0000 0000$_H$	32	FFFF ADAC$_H$	0	32
DMASS	DTS Source Address Register	DTSA119	XXXX XXXX$_H$	32	FFFF ADC0$_H$	0	32
DMASS	DTS Destination Address Register	DTDA119	XXXX XXXX$_H$	32	FFFF ADC4$_H$	0	32
DMASS	DTS Transfer Count Register	DTTC119	XXXX XXXX$_H$	32	FFFF ADC8$_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT119	XXXX XXXX$_H$	32	FFFF ADCC$_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA119	XXXX XXXX$_H$	32	FFFF ADD0$_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA119	XXXX XXXX$_H$	32	FFFF ADD4$_H$	0	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
DMASS	DTS Reload Transfer Count Register	DTRTC119	XXXX $XXXX_H$	32	FFFF $ADD8_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC119	XXXX $XXXX_H$	32	FFFF $ADDC_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL119	0000 0000_H	32	FFFF $ADE0_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST119	0000 0000_H	32	FFFF $ADE4_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS119	0000 0000_H	32	FFFF $ADE8_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC119	0000 0000_H	32	FFFF $ADEC_H$	0	32
DMASS	DTS Source Address Register	DTSA120	XXXX $XXXX_H$	32	FFFF $AE00_H$	0	32
DMASS	DTS Destination Address Register	DTDA120	XXXX $XXXX_H$	32	FFFF $AE04_H$	0	32
DMASS	DTS Transfer Count Register	DTTC120	XXXX $XXXX_H$	32	FFFF $AE08_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT120	XXXX $XXXX_H$	32	FFFF $AE0C_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA120	XXXX $XXXX_H$	32	FFFF $AE10_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA120	XXXX $XXXX_H$	32	FFFF $AE14_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC120	XXXX $XXXX_H$	32	FFFF $AE18_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC120	XXXX $XXXX_H$	32	FFFF $AE1C_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL120	0000 0000_H	32	FFFF $AE20_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST120	0000 0000_H	32	FFFF $AE24_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS120	0000 0000_H	32	FFFF $AE28_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC120	0000 0000_H	32	FFFF $AE2C_H$	0	32
DMASS	DTS Source Address Register	DTSA121	XXXX $XXXX_H$	32	FFFF $AE40_H$	0	32
DMASS	DTS Destination Address Register	DTDA121	XXXX $XXXX_H$	32	FFFF $AE44_H$	0	32
DMASS	DTS Transfer Count Register	DTTC121	XXXX $XXXX_H$	32	FFFF $AE48_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT121	XXXX $XXXX_H$	32	FFFF $AE4C_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA121	XXXX $XXXX_H$	32	FFFF $AE50_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA121	XXXX $XXXX_H$	32	FFFF $AE54_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC121	XXXX $XXXX_H$	32	FFFF $AE58_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC121	XXXX $XXXX_H$	32	FFFF $AE5C_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL121	0000 0000_H	32	FFFF $AE60_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST121	0000 0000_H	32	FFFF $AE64_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS121	0000 0000_H	32	FFFF $AE68_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC121	0000 0000_H	32	FFFF $AE6C_H$	0	32
DMASS	DTS Source Address Register	DTSA122	XXXX $XXXX_H$	32	FFFF $AE80_H$	0	32
DMASS	DTS Destination Address Register	DTDA122	XXXX $XXXX_H$	32	FFFF $AE84_H$	0	32
DMASS	DTS Transfer Count Register	DTTC122	XXXX $XXXX_H$	32	FFFF $AE88_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT122	XXXX $XXXX_H$	32	FFFF $AE8C_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA122	XXXX $XXXX_H$	32	FFFF $AE90_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA122	XXXX $XXXX_H$	32	FFFF $AE94_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC122	XXXX $XXXX_H$	32	FFFF $AE98_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC122	XXXX $XXXX_H$	32	FFFF $AE9C_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL122	0000 0000_H	32	FFFF $AEA0_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST122	0000 0000_H	32	FFFF $AEA4_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS122	0000 0000_H	32	FFFF $AEA8_H$	0	32

(181/201)

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
DMASS	DTSFSL Transfer Request Clear Register	DTFSC122	0000 0000_H	32	FFFF $AEAC_H$	0	32
DMASS	DTS Source Address Register	DTSA123	XXXX $XXXX_H$	32	FFFF $AEC0_H$	0	32
DMASS	DTS Destination Address Register	DTDA123	XXXX $XXXX_H$	32	FFFF $AEC4_H$	0	32
DMASS	DTS Transfer Count Register	DTTC123	XXXX $XXXX_H$	32	FFFF $AEC8_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT123	XXXX $XXXX_H$	32	FFFF $AECC_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA123	XXXX $XXXX_H$	32	FFFF $AED0_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA123	XXXX $XXXX_H$	32	FFFF $AED4_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC123	XXXX $XXXX_H$	32	FFFF $AED8_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC123	XXXX $XXXX_H$	32	FFFF $AEDC_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL123	0000 0000_H	32	FFFF $AEE0_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST123	0000 0000_H	32	FFFF $AEE4_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS123	0000 0000_H	32	FFFF $AEE8_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC123	0000 0000_H	32	FFFF $AEEC_H$	0	32
DMASS	DTS Source Address Register	DTSA124	XXXX $XXXX_H$	32	FFFF $AF00_H$	0	32
DMASS	DTS Destination Address Register	DTDA124	XXXX $XXXX_H$	32	FFFF $AF04_H$	0	32
DMASS	DTS Transfer Count Register	DTTC124	XXXX $XXXX_H$	32	FFFF $AF08_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT124	XXXX $XXXX_H$	32	FFFF $AF0C_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA124	XXXX $XXXX_H$	32	FFFF $AF10_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA124	XXXX $XXXX_H$	32	FFFF $AF14_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC124	XXXX $XXXX_H$	32	FFFF $AF18_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC124	XXXX $XXXX_H$	32	FFFF $AF1C_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL124	0000 0000_H	32	FFFF $AF20_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST124	0000 0000_H	32	FFFF $AF24_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS124	0000 0000_H	32	FFFF $AF28_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC124	0000 0000_H	32	FFFF $AF2C_H$	0	32
DMASS	DTS Source Address Register	DTSA125	XXXX $XXXX_H$	32	FFFF $AF40_H$	0	32
DMASS	DTS Destination Address Register	DTDA125	XXXX $XXXX_H$	32	FFFF $AF44_H$	0	32
DMASS	DTS Transfer Count Register	DTTC125	XXXX $XXXX_H$	32	FFFF $AF48_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT125	XXXX $XXXX_H$	32	FFFF $AF4C_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA125	XXXX $XXXX_H$	32	FFFF $AF50_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA125	XXXX $XXXX_H$	32	FFFF $AF54_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC125	XXXX $XXXX_H$	32	FFFF $AF58_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC125	XXXX $XXXX_H$	32	FFFF $AF5C_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL125	0000 0000_H	32	FFFF $AF60_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST125	0000 0000_H	32	FFFF $AF64_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS125	0000 0000_H	32	FFFF $AF68_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC125	0000 0000_H	32	FFFF $AF6C_H$	0	32
DMASS	DTS Source Address Register	DTSA126	XXXX $XXXX_H$	32	FFFF $AF80_H$	0	32
DMASS	DTS Destination Address Register	DTDA126	XXXX $XXXX_H$	32	FFFF $AF84_H$	0	32
DMASS	DTS Transfer Count Register	DTTC126	XXXX $XXXX_H$	32	FFFF $AF88_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT126	XXXX $XXXX_H$	32	FFFF $AF8C_H$	0	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
DMASS	DTS Reload Source Address Register	DTRSA126	XXXX XXXX$_H$	32	FFFF AF90$_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA126	XXXX XXXX$_H$	32	FFFF AF94$_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC126	XXXX XXXX$_H$	32	FFFF AF98$_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC126	XXXX XXXX$_H$	32	FFFF AF9C$_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL126	0000 0000$_H$	32	FFFF AFA0$_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST126	0000 0000$_H$	32	FFFF AFA4$_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS126	0000 0000$_H$	32	FFFF AFA8$_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC126	0000 0000$_H$	32	FFFF AFAC$_H$	0	32
DMASS	DTS Source Address Register	DTSA127	XXXX XXXX$_H$	32	FFFF AFC0$_H$	0	32
DMASS	DTS Destination Address Register	DTDA127	XXXX XXXX$_H$	32	FFFF AFC4$_H$	0	32
DMASS	DTS Transfer Count Register	DTTC127	XXXX XXXX$_H$	32	FFFF AFC8$_H$	0	32
DMASS	DTS Transfer Control Register	DTTCT127	XXXX XXXX$_H$	32	FFFF AFCC$_H$	0	32
DMASS	DTS Reload Source Address Register	DTRSA127	XXXX XXXX$_H$	32	FFFF AFD0$_H$	0	32
DMASS	DTS Reload Destination Address Register	DTRDA127	XXXX XXXX$_H$	32	FFFF AFD4$_H$	0	32
DMASS	DTS Reload Transfer Count Register	DTRTC127	XXXX XXXX$_H$	32	FFFF AFD8$_H$	0	32
DMASS	DTS Transfer Count Compare Register	DTTCC127	XXXX XXXX$_H$	32	FFFF AFDC$_H$	0	32
DMASS	DTSFSL Operation Setting Register	DTFSL127	0000 0000$_H$	32	FFFF AFE0$_H$	0	32
DMASS	DTSFSL Transfer Request Status Register	DTFST127	0000 0000$_H$	32	FFFF AFE4$_H$	0	32
DMASS	DTSFSL Transfer Request set Register	DTFSS127	0000 0000$_H$	32	FFFF AFE8$_H$	0	32
DMASS	DTSFSL Transfer Request Clear Register	DTFSC127	0000 0000$_H$	32	FFFF AFEC$_H$	0	32
INTC2	EI Level Interrupt Control Register 33	EIC33	008F$_H$	16	FFFF B042$_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 38	EIC38	008F$_H$	16	FFFF B04C$_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 39	EIC39	008F$_H$	16	FFFF B04E$_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 40	EIC40	008F$_H$	16	FFFF B050$_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 41	EIC41	008F$_H$	16	FFFF B052$_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 42	EIC42	008F$_H$	16	FFFF B054$_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 43	EIC43	008F$_H$	16	FFFF B056$_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 44	EIC44	008F$_H$	16	FFFF B058$_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 45	EIC45	008F$_H$	16	FFFF B05A$_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 46	EIC46	008F$_H$	16	FFFF B05C$_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 47	EIC47	008F$_H$	16	FFFF B05E$_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 48	EIC48	008F$_H$	16	FFFF B060$_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 49	EIC49	008F$_H$	16	FFFF B062$_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 50	EIC50	008F$_H$	16	FFFF B064$_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 51	EIC51	008F$_H$	16	FFFF B066$_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 52	EIC52	008F$_H$	16	FFFF B068$_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 53	EIC53	008F$_H$	16	FFFF B06A$_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 54	EIC54	008F$_H$	16	FFFF B06C$_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 55	EIC55	008F$_H$	16	FFFF B06E$_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 57	EIC57	008F$_H$	16	FFFF B072$_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 58	EIC58	008F$_H$	16	FFFF B074$_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 59	EIC59	008F$_H$	16	FFFF B076$_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 60	EIC60	008F$_H$	16	FFFF B078$_H$	0	8, 16

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
INTC2	EI Level Interrupt Control Register 61	EIC61	$008F_H$	16	FFFF $B07A_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 62	EIC62	$008F_H$	16	FFFF $B07C_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 63	EIC63	$008F_H$	16	FFFF $B07E_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 64	EIC64	$008F_H$	16	FFFF $B080_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 65	EIC65	$008F_H$	16	FFFF $B082_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 66	EIC66	$008F_H$	16	FFFF $B084_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 67	EIC67	$008F_H$	16	FFFF $B086_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 68	EIC68	$008F_H$	16	FFFF $B088_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 69	EIC69	$008F_H$	16	FFFF $B08A_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 70	EIC70	$008F_H$	16	FFFF $B08C_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 71	EIC71	$008F_H$	16	FFFF $B08E_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 72	EIC72	$008F_H$	16	FFFF $B090_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 73	EIC73	$008F_H$	16	FFFF $B092_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 74	EIC74	$008F_H$	16	FFFF $B094_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 75	EIC75	$008F_H$	16	FFFF $B096_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 76	EIC76	$008F_H$	16	FFFF $B098_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 77	EIC77	$008F_H$	16	FFFF $B09A_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 78	EIC78	$008F_H$	16	FFFF $B09C_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 79	EIC79	$008F_H$	16	FFFF $B09E_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 80	EIC80	$008F_H$	16	FFFF $B0A0_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 81	EIC81	$008F_H$	16	FFFF $B0A2_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 82	EIC82	$008F_H$	16	FFFF $B0A4_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 83	EIC83	$008F_H$	16	FFFF $B0A6_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 84	EIC84	$008F_H$	16	FFFF $B0A8_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 85	EIC85	$008F_H$	16	FFFF $B0AA_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 86	EIC86	$008F_H$	16	FFFF $B0AC_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 87	EIC87	$008F_H$	16	FFFF $B0AE_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 88	EIC88	$008F_H$	16	FFFF $B0B0_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 89	EIC89	$008F_H$	16	FFFF $B0B2_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 90	EIC90	$008F_H$	16	FFFF $B0B4_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 91	EIC91	$008F_H$	16	FFFF $B0B6_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 92	EIC92	$008F_H$	16	FFFF $B0B8_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 93	EIC93	$008F_H$	16	FFFF $B0BA_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 94	EIC94	$008F_H$	16	FFFF $B0BC_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 95	EIC95	$008F_H$	16	FFFF $B0BE_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 96	EIC96	$008F_H$	16	FFFF $B0C0_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 97	EIC97	$008F_H$	16	FFFF $B0C2_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 98	EIC98	$008F_H$	16	FFFF $B0C4_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 99	EIC99	$008F_H$	16	FFFF $B0C6_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 100	EIC100	$008F_H$	16	FFFF $B0C8_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 101	EIC101	$008F_H$	16	FFFF $B0CA_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 102	EIC102	$008F_H$	16	FFFF $B0CC_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 103	EIC103	$008F_H$	16	FFFF $B0CE_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 104	EIC104	$008F_H$	16	FFFF $B0D0_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 105	EIC105	$008F_H$	16	FFFF $B0D2_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 106	EIC106	$008F_H$	16	FFFF $B0D4_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 107	EIC107	$008F_H$	16	FFFF $B0D6_H$	0	8, 16

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
INTC2	EI Level Interrupt Control Register 108	EIC108	$008F_H$	16	FFFF $B0D8_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 109	EIC109	$008F_H$	16	FFFF $B0DA_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 110	EIC110	$008F_H$	16	FFFF $B0DC_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 111	EIC111	$008F_H$	16	FFFF $B0DE_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 112	EIC112	$008F_H$	16	FFFF $B0E0_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 113	EIC113	$008F_H$	16	FFFF $B0E2_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 114	EIC114	$008F_H$	16	FFFF $B0E4_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 115	EIC115	$008F_H$	16	FFFF $B0E6_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 116	EIC116	$008F_H$	16	FFFF $B0E8_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 117	EIC117	$008F_H$	16	FFFF $B0EA_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 118	EIC118	$008F_H$	16	FFFF $B0EC_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 119	EIC119	$008F_H$	16	FFFF $B0EE_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 120	EIC120	$008F_H$	16	FFFF $B0F0_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 121	EIC121	$008F_H$	16	FFFF $B0F2_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 122	EIC122	$008F_H$	16	FFFF $B0F4_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 123	EIC123	$008F_H$	16	FFFF $B0F6_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 124	EIC124	$008F_H$	16	FFFF $B0F8_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 125	EIC125	$008F_H$	16	FFFF $B0FA_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 126	EIC126	$008F_H$	16	FFFF $B0FC_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 127	EIC127	$008F_H$	16	FFFF $B0FE_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 128	EIC128	$008F_H$	16	FFFF $B100_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 129	EIC129	$008F_H$	16	FFFF $B102_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 130	EIC130	$008F_H$	16	FFFF $B104_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 131	EIC131	$008F_H$	16	FFFF $B106_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 132	EIC132	$008F_H$	16	FFFF $B108_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 133	EIC133	$008F_H$	16	FFFF $B10A_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 134	EIC134	$008F_H$	16	FFFF $B10C_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 135	EIC135	$008F_H$	16	FFFF $B10E_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 136	EIC136	$008F_H$	16	FFFF $B110_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 137	EIC137	$008F_H$	16	FFFF $B112_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 138	EIC138	$008F_H$	16	FFFF $B114_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 139	EIC139	$008F_H$	16	FFFF $B116_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 140	EIC140	$008F_H$	16	FFFF $B118_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 141	EIC141	$008F_H$	16	FFFF $B11A_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 142	EIC142	$008F_H$	16	FFFF $B11C_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 143	EIC143	$008F_H$	16	FFFF $B11E_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 144	EIC144	$008F_H$	16	FFFF $B120_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 145	EIC145	$008F_H$	16	FFFF $B122_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 146	EIC146	$008F_H$	16	FFFF $B124_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 147	EIC147	$008F_H$	16	FFFF $B126_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 148	EIC148	$008F_H$	16	FFFF $B128_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 149	EIC149	$008F_H$	16	FFFF $B12A_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 150	EIC150	$008F_H$	16	FFFF $B12C_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 151	EIC151	$008F_H$	16	FFFF $B12E_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 152	EIC152	$008F_H$	16	FFFF $B130_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 153	EIC153	$008F_H$	16	FFFF $B132_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 154	EIC154	$008F_H$	16	FFFF $B134_H$	0	8, 16

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
INTC2	EI Level Interrupt Control Register 155	EIC155	$008F_H$	16	FFFF $B136_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 156	EIC156	$008F_H$	16	FFFF $B138_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 157	EIC157	$008F_H$	16	FFFF $B13A_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 158	EIC158	$008F_H$	16	FFFF $B13C_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 159	EIC159	$008F_H$	16	FFFF $B13E_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 160	EIC160	$008F_H$	16	FFFF $B140_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 161	EIC161	$008F_H$	16	FFFF $B142_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 162	EIC162	$008F_H$	16	FFFF $B144_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 163	EIC163	$008F_H$	16	FFFF $B146_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 164	EIC164	$008F_H$	16	FFFF $B148_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 165	EIC165	$008F_H$	16	FFFF $B14A_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 166	EIC166	$008F_H$	16	FFFF $B14C_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 167	EIC167	$008F_H$	16	FFFF $B14E_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 168	EIC168	$008F_H$	16	FFFF $B150_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 169	EIC169	$008F_H$	16	FFFF $B152_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 170	EIC170	$008F_H$	16	FFFF $B154_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 171	EIC171	$008F_H$	16	FFFF $B156_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 172	EIC172	$008F_H$	16	FFFF $B158_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 173	EIC173	$008F_H$	16	FFFF $B15A_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 174	EIC174	$008F_H$	16	FFFF $B15C_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 175	EIC175	$008F_H$	16	FFFF $B15E_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 176	EIC176	$008F_H$	16	FFFF $B160_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 177	EIC177	$008F_H$	16	FFFF $B162_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 178	EIC178	$008F_H$	16	FFFF $B164_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 179	EIC179	$008F_H$	16	FFFF $B166_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 180	EIC180	$008F_H$	16	FFFF $B168_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 181	EIC181	$008F_H$	16	FFFF $B16A_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 182	EIC182	$008F_H$	16	FFFF $B16C_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 183	EIC183	$008F_H$	16	FFFF $B16E_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 184	EIC184	$008F_H$	16	FFFF $B170_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 185	EIC185	$008F_H$	16	FFFF $B172_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 186	EIC186	$008F_H$	16	FFFF $B174_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 187	EIC187	$008F_H$	16	FFFF $B176_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 188	EIC188	$008F_H$	16	FFFF $B178_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 189	EIC189	$008F_H$	16	FFFF $B17A_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 190	EIC190	$008F_H$	16	FFFF $B17C_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 191	EIC191	$008F_H$	16	FFFF $B17E_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 192	EIC192	$008F_H$	16	FFFF $B180_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 193	EIC193	$008F_H$	16	FFFF $B182_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 194	EIC194	$008F_H$	16	FFFF $B184_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 195	EIC195	$008F_H$	16	FFFF $B186_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 196	EIC196	$008F_H$	16	FFFF $B188_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 197	EIC197	$008F_H$	16	FFFF $B18A_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 198	EIC198	$008F_H$	16	FFFF $B18C_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 199	EIC199	$008F_H$	16	FFFF $B18E_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 200	EIC200	$008F_H$	16	FFFF $B190_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 201	EIC201	$008F_H$	16	FFFF $B192_H$	0	8, 16

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
INTC2	EI Level Interrupt Control Register 202	EIC202	$008F_H$	16	FFFF $B194_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 203	EIC203	$008F_H$	16	FFFF $B196_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 204	EIC204	$008F_H$	16	FFFF $B198_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 205	EIC205	$008F_H$	16	FFFF $B19A_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 206	EIC206	$008F_H$	16	FFFF $B19C_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 207	EIC207	$008F_H$	16	FFFF $B19E_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 208	EIC208	$008F_H$	16	FFFF $B1A0_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 209	EIC209	$008F_H$	16	FFFF $B1A2_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 210	EIC210	$008F_H$	16	FFFF $B1A4_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 211	EIC211	$008F_H$	16	FFFF $B1A6_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 212	EIC212	$008F_H$	16	FFFF $B1A8_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 213	EIC213	$008F_H$	16	FFFF $B1AA_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 214	EIC214	$008F_H$	16	FFFF $B1AC_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 215	EIC215	$008F_H$	16	FFFF $B1AE_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 216	EIC216	$008F_H$	16	FFFF $B1B0_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 217	EIC217	$008F_H$	16	FFFF $B1B2_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 218	EIC218	$008F_H$	16	FFFF $B1B4_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 219	EIC219	$008F_H$	16	FFFF $B1B6_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 220	EIC220	$008F_H$	16	FFFF $B1B8_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 221	EIC221	$008F_H$	16	FFFF $B1BA_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 222	EIC222	$008F_H$	16	FFFF $B1BC_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 223	EIC223	$008F_H$	16	FFFF $B1BE_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 224	EIC224	$008F_H$	16	FFFF $B1C0_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 225	EIC225	$008F_H$	16	FFFF $B1C2_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 226	EIC226	$008F_H$	16	FFFF $B1C4_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 227	EIC227	$008F_H$	16	FFFF $B1C6_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 228	EIC228	$008F_H$	16	FFFF $B1C8_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 229	EIC229	$008F_H$	16	FFFF $B1CA_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 230	EIC230	$008F_H$	16	FFFF $B1CC_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 231	EIC231	$008F_H$	16	FFFF $B1CE_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 232	EIC232	$008F_H$	16	FFFF $B1D0_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 233	EIC233	$008F_H$	16	FFFF $B1D2_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 234	EIC234	$008F_H$	16	FFFF $B1D4_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 235	EIC235	$008F_H$	16	FFFF $B1D6_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 236	EIC236	$008F_H$	16	FFFF $B1D8_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 237	EIC237	$008F_H$	16	FFFF $B1DA_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 238	EIC238	$008F_H$	16	FFFF $B1DC_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 239	EIC239	$008F_H$	16	FFFF $B1DE_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 240	EIC240	$008F_H$	16	FFFF $B1E0_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 241	EIC241	$008F_H$	16	FFFF $B1E2_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 242	EIC242	$808F_H$	16	FFFF $B1E4_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 243	EIC243	$808F_H$	16	FFFF $B1E6_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 244	EIC244	$808F_H$	16	FFFF $B1E8_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 245	EIC245	$808F_H$	16	FFFF $B1EA_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 246	EIC246	$808F_H$	16	FFFF $B1EC_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 247	EIC247	$808F_H$	16	FFFF $B1EE_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 248	EIC248	$808F_H$	16	FFFF $B1F0_H$	0	8, 16

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
INTC2	EI Level Interrupt Control Register 249	EIC249	$808F_H$	16	FFFF $B1F2_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 250	EIC250	$808F_H$	16	FFFF $B1F4_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 251	EIC251	$808F_H$	16	FFFF $B1F6_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 252	EIC252	$808F_H$	16	FFFF $B1F8_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 253	EIC253	$808F_H$	16	FFFF $B1FA_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 254	EIC254	$808F_H$	16	FFFF $B1FC_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 255	EIC255	$808F_H$	16	FFFF $B1FE_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 256	EIC256	$808F_H$	16	FFFF $B200_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 257	EIC257	$808F_H$	16	FFFF $B202_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 258	EIC258	$808F_H$	16	FFFF $B204_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 259	EIC259	$808F_H$	16	FFFF $B206_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 260	EIC260	$808F_H$	16	FFFF $B208_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 261	EIC261	$808F_H$	16	FFFF $B20A_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 262	EIC262	$808F_H$	16	FFFF $B20C_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 263	EIC263	$808F_H$	16	FFFF $B20E_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 264	EIC264	$808F_H$	16	FFFF $B210_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 265	EIC265	$808F_H$	16	FFFF $B212_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 266	EIC266	$808F_H$	16	FFFF $B214_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 267	EIC267	$808F_H$	16	FFFF $B216_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 268	EIC268	$808F_H$	16	FFFF $B218_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 269	EIC269	$808F_H$	16	FFFF $B21A_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 270	EIC270	$808F_H$	16	FFFF $B21C_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 271	EIC271	$808F_H$	16	FFFF $B21E_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 272	EIC272	$808F_H$	16	FFFF $B220_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 273	EIC273	$808F_H$	16	FFFF $B222_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 274	EIC274	$808F_H$	16	FFFF $B224_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 275	EIC275	$808F_H$	16	FFFF $B226_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 276	EIC276	$808F_H$	16	FFFF $B228_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 277	EIC277	$808F_H$	16	FFFF $B22A_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 278	EIC278	$808F_H$	16	FFFF $B22C_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 279	EIC279	$808F_H$	16	FFFF $B22E_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 280	EIC280	$808F_H$	16	FFFF $B230_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 281	EIC281	$808F_H$	16	FFFF $B232_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 282	EIC282	$808F_H$	16	FFFF $B234_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 283	EIC283	$808F_H$	16	FFFF $B236_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 284	EIC284	$808F_H$	16	FFFF $B238_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 285	EIC285	$808F_H$	16	FFFF $B23A_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 286	EIC286	$008F_H$	16	FFFF $B23C_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 287	EIC287	$008F_H$	16	FFFF $B23E_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 288	EIC288	$008F_H$	16	FFFF $B240_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 289	EIC289	$008F_H$	16	FFFF $B242_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 290	EIC290	$008F_H$	16	FFFF $B244_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 291	EIC291	$008F_H$	16	FFFF $B246_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 292	EIC292	$008F_H$	16	FFFF $B248_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 293	EIC293	$008F_H$	16	FFFF $B24A_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 294	EIC294	$008F_H$	16	FFFF $B24C_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 295	EIC295	$008F_H$	16	FFFF $B24E_H$	0	8, 16

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
INTC2	EI Level Interrupt Control Register 296	EIC296	$008F_H$	16	FFFF $B250_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 297	EIC297	$008F_H$	16	FFFF $B252_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 298	EIC298	$008F_H$	16	FFFF $B254_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 299	EIC299	$008F_H$	16	FFFF $B256_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 300	EIC300	$008F_H$	16	FFFF $B258_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 301	EIC301	$008F_H$	16	FFFF $B25A_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 302	EIC302	$008F_H$	16	FFFF $B25C_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 303	EIC303	$008F_H$	16	FFFF $B25E_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 304	EIC304	$008F_H$	16	FFFF $B260_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 305	EIC305	$008F_H$	16	FFFF $B262_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 306	EIC306	$008F_H$	16	FFFF $B264_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 307	EIC307	$008F_H$	16	FFFF $B266_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 308	EIC308	$008F_H$	16	FFFF $B268_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 309	EIC309	$008F_H$	16	FFFF $B26A_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 310	EIC310	$008F_H$	16	FFFF $B26C_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 311	EIC311	$008F_H$	16	FFFF $B26E_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 312	EIC312	$008F_H$	16	FFFF $B270_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 313	EIC313	$008F_H$	16	FFFF $B272_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 314	EIC314	$008F_H$	16	FFFF $B274_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 315	EIC315	$008F_H$	16	FFFF $B276_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 316	EIC316	$008F_H$	16	FFFF $B278_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 317	EIC317	$008F_H$	16	FFFF $B27A_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 318	EIC318	$008F_H$	16	FFFF $B27C_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 321	EIC321	$008F_H$	16	FFFF $B282_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 322	EIC322	$008F_H$	16	FFFF $B284_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 323	EIC323	$008F_H$	16	FFFF $B286_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 324	EIC324	$008F_H$	16	FFFF $B288_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 325	EIC325	$008F_H$	16	FFFF $B28A_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 326	EIC326	$008F_H$	16	FFFF $B28C_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 329	EIC329	$008F_H$	16	FFFF $B292_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 330	EIC330	$008F_H$	16	FFFF $B294_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 331	EIC331	$008F_H$	16	FFFF $B296_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 332	EIC332	$008F_H$	16	FFFF $B298_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 333	EIC333	$008F_H$	16	FFFF $B29A_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 334	EIC334	$008F_H$	16	FFFF $B29C_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 337	EIC337	$008F_H$	16	FFFF $B2A2_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 338	EIC338	$008F_H$	16	FFFF $B2A4_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 339	EIC339	$008F_H$	16	FFFF $B2A6_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 340	EIC340	$008F_H$	16	FFFF $B2A8_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 341	EIC341	$008F_H$	16	FFFF $B2AA_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 342	EIC342	$008F_H$	16	FFFF $B2AC_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 343	EIC343	$008F_H$	16	FFFF $B2AE_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 344	EIC344	$008F_H$	16	FFFF $B2B0_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 345	EIC345	$008F_H$	16	FFFF $B2B2_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 346	EIC346	$008F_H$	16	FFFF $B2B4_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 347	EIC347	$008F_H$	16	FFFF $B2B6_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 348	EIC348	$008F_H$	16	FFFF $B2B8_H$	0	8, 16

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
INTC2	EI Level Interrupt Control Register 349	EIC349	$008F_H$	16	FFFF $B2BA_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 350	EIC350	$008F_H$	16	FFFF $B2BC_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 351	EIC351	$008F_H$	16	FFFF $B2BE_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 352	EIC352	$008F_H$	16	FFFF $B2C0_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 353	EIC353	$008F_H$	16	FFFF $B2C2_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 354	EIC354	$008F_H$	16	FFFF $B2C4_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 355	EIC355	$008F_H$	16	FFFF $B2C6_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 356	EIC356	$008F_H$	16	FFFF $B2C8_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 357	EIC357	$008F_H$	16	FFFF $B2CA_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 358	EIC358	$008F_H$	16	FFFF $B2CC_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 359	EIC359	$008F_H$	16	FFFF $B2CE_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 360	EIC360	$008F_H$	16	FFFF $B2D0_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 361	EIC361	$008F_H$	16	FFFF $B2D2_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 362	EIC362	$008F_H$	16	FFFF $B2D4_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 363	EIC363	$008F_H$	16	FFFF $B2D6_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 364	EIC364	$008F_H$	16	FFFF $B2D8_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 365	EIC365	$008F_H$	16	FFFF $B2DA_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 366	EIC366	$008F_H$	16	FFFF $B2DC_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 367	EIC367	$008F_H$	16	FFFF $B2DE_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 368	EIC368	$008F_H$	16	FFFF $B2E0_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 369	EIC369	$008F_H$	16	FFFF $B2E2_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 370	EIC370	$008F_H$	16	FFFF $B2E4_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 371	EIC371	$008F_H$	16	FFFF $B2E6_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 372	EIC372	$008F_H$	16	FFFF $B2E8_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 373	EIC373	$008F_H$	16	FFFF $B2EA_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 374	EIC374	$008F_H$	16	FFFF $B2EC_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 375	EIC375	$008F_H$	16	FFFF $B2EE_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 376	EIC376	$008F_H$	16	FFFF $B2F0_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 377	EIC377	$008F_H$	16	FFFF $B2F2_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 378	EIC378	$008F_H$	16	FFFF $B2F4_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 379	EIC379	$008F_H$	16	FFFF $B2F6_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 380	EIC380	$008F_H$	16	FFFF $B2F8_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 381	EIC381	$008F_H$	16	FFFF $B2FA_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 382	EIC382	$008F_H$	16	FFFF $B2FC_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 383	EIC383	$008F_H$	16	FFFF $B2FE_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 384	EIC384	$008F_H$	16	FFFF $B300_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 385	EIC385	$008F_H$	16	FFFF $B302_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 386	EIC386	$008F_H$	16	FFFF $B304_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 387	EIC387	$008F_H$	16	FFFF $B306_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 388	EIC388	$008F_H$	16	FFFF $B308_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 389	EIC389	$008F_H$	16	FFFF $B30A_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 390	EIC390	$008F_H$	16	FFFF $B30C_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 391	EIC391	$008F_H$	16	FFFF $B30E_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 392	EIC392	$008F_H$	16	FFFF $B310_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 393	EIC393	$008F_H$	16	FFFF $B312_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 394	EIC394	$008F_H$	16	FFFF $B314_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 395	EIC395	$008F_H$	16	FFFF $B316_H$	0	8, 16

(190/201)

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
INTC2	EI Level Interrupt Control Register 396	EIC396	$008F_H$	16	FFFF $B318_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 397	EIC397	$008F_H$	16	FFFF $B31A_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 398	EIC398	$008F_H$	16	FFFF $B31C_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 399	EIC399	$008F_H$	16	FFFF $B31E_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 400	EIC400	$008F_H$	16	FFFF $B320_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 401	EIC401	$808F_H$	16	FFFF $B322_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 402	EIC402	$808F_H$	16	FFFF $B324_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 403	EIC403	$808F_H$	16	FFFF $B326_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 404	EIC404	$808F_H$	16	FFFF $B328_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 405	EIC405	$808F_H$	16	FFFF $B32A_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 406	EIC406	$808F_H$	16	FFFF $B32C_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 407	EIC407	$808F_H$	16	FFFF $B32E_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 408	EIC408	$808F_H$	16	FFFF $B330_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 409	EIC409	$808F_H$	16	FFFF $B332_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 410	EIC410	$808F_H$	16	FFFF $B334_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 411	EIC411	$808F_H$	16	FFFF $B336_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 412	EIC412	$808F_H$	16	FFFF $B338_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 413	EIC413	$808F_H$	16	FFFF $B33A_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 414	EIC414	$808F_H$	16	FFFF $B33C_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 415	EIC415	$808F_H$	16	FFFF $B33E_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 416	EIC416	$808F_H$	16	FFFF $B340_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 417	EIC417	$808F_H$	16	FFFF $B342_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 418	EIC418	$808F_H$	16	FFFF $B344_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 419	EIC419	$808F_H$	16	FFFF $B346_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 420	EIC420	$808F_H$	16	FFFF $B348_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 421	EIC421	$808F_H$	16	FFFF $B34A_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 422	EIC422	$808F_H$	16	FFFF $B34C_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 423	EIC423	$808F_H$	16	FFFF $B34E_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 424	EIC424	$808F_H$	16	FFFF $B350_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 425	EIC425	$808F_H$	16	FFFF $B352_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 429	EIC429	$808F_H$	16	FFFF $B35A_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 434	EIC434	$808F_H$	16	FFFF $B364_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 435	EIC435	$808F_H$	16	FFFF $B366_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 436	EIC436	$808F_H$	16	FFFF $B368_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 437	EIC437	$808F_H$	16	FFFF $B36A_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 438	EIC438	$808F_H$	16	FFFF $B36C_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 439	EIC439	$808F_H$	16	FFFF $B36E_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 440	EIC440	$808F_H$	16	FFFF $B370_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 441	EIC441	$808F_H$	16	FFFF $B372_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 442	EIC442	$808F_H$	16	FFFF $B374_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 443	EIC443	$808F_H$	16	FFFF $B376_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 444	EIC444	$808F_H$	16	FFFF $B378_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 445	EIC445	$008F_H$	16	FFFF $B37A_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 446	EIC446	$008F_H$	16	FFFF $B37C_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 447	EIC447	$808F_H$	16	FFFF $B37E_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 448	EIC448	$808F_H$	16	FFFF $B380_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 449	EIC449	$008F_H$	16	FFFF $B382_H$	0	8, 16

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
INTC2	EI Level Interrupt Control Register 450	EIC450	$008F_H$	16	FFFF $B384_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 451	EIC451	$808F_H$	16	FFFF $B386_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 452	EIC452	$808F_H$	16	FFFF $B388_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 453	EIC453	$008F_H$	16	FFFF $B38A_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 454	EIC454	$008F_H$	16	FFFF $B38C_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 455	EIC455	$808F_H$	16	FFFF $B38E_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 456	EIC456	$808F_H$	16	FFFF $B390_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 457	EIC457	$008F_H$	16	FFFF $B392_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 458	EIC458	$008F_H$	16	FFFF $B394_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 459	EIC459	$808F_H$	16	FFFF $B396_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 460	EIC460	$008F_H$	16	FFFF $B398_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 461	EIC461	$008F_H$	16	FFFF $B39A_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 462	EIC462	$008F_H$	16	FFFF $B39C_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 463	EIC463	$008F_H$	16	FFFF $B39E_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 464	EIC464	$008F_H$	16	FFFF $B3A0_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 465	EIC465	$008F_H$	16	FFFF $B3A2_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 466	EIC466	$008F_H$	16	FFFF $B3A4_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 467	EIC467	$008F_H$	16	FFFF $B3A6_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 468	EIC468	$008F_H$	16	FFFF $B3A8_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 469	EIC469	$008F_H$	16	FFFF $B3AA_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 470	EIC470	$008F_H$	16	FFFF $B3AC_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 471	EIC471	$008F_H$	16	FFFF $B3AE_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 472	EIC472	$008F_H$	16	FFFF $B3B0_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 473	EIC473	$008F_H$	16	FFFF $B3B2_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 474	EIC474	$008F_H$	16	FFFF $B3B4_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 475	EIC475	$008F_H$	16	FFFF $B3B6_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 480	EIC480	$808F_H$	16	FFFF $B3C0_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 481	EIC481	$808F_H$	16	FFFF $B3C2_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 482	EIC482	$808F_H$	16	FFFF $B3C4_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 483	EIC483	$808F_H$	16	FFFF $B3C6_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 484	EIC484	$808F_H$	16	FFFF $B3C8_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 485	EIC485	$808F_H$	16	FFFF $B3CA_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 486	EIC486	$808F_H$	16	FFFF $B3CC_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 487	EIC487	$808F_H$	16	FFFF $B3CE_H$	0	8, 16
INTC2	EI Level Interrupt Control Register 489	EIC489	$008F_H$	16	FFFF $B3D2_H$	0	8, 16
INTC2	EI Level Interrupt Mask Register 1	IMR1	$FFFF FFFF_H$	32	FFFF $B404_H$	0	8, 16, 32
INTC2	EI Level Interrupt Mask Register 2	IMR2	$FFFF FFFF_H$	32	FFFF $B408_H$	0	8, 16, 32
INTC2	EI Level Interrupt Mask Register 3	IMR3	$FFFF FFFF_H$	32	FFFF $B40C_H$	0	8, 16, 32
INTC2	EI Level Interrupt Mask Register 4	IMR4	$FFFF FFFF_H$	32	FFFF $B410_H$	0	8, 16, 32
INTC2	EI Level Interrupt Mask Register 5	IMR5	$FFFF FFFF_H$	32	FFFF $B414_H$	0	8, 16, 32
INTC2	EI Level Interrupt Mask Register 6	IMR6	$FFFF FFFF_H$	32	FFFF $B418_H$	0	8, 16, 32
INTC2	EI Level Interrupt Mask Register 7	IMR7	$FFFF FFFF_H$	32	FFFF $B41C_H$	0	8, 16, 32
INTC2	EI Level Interrupt Mask Register 8	IMR8	$FFFF FFFF_H$	32	FFFF $B420_H$	0	8, 16, 32
INTC2	EI Level Interrupt Mask Register 9	IMR9	$FFFF FFFF_H$	32	FFFF $B424_H$	0	8, 16, 32
INTC2	EI Level Interrupt Mask Register 10	IMR10	$FFFF FFFF_H$	32	FFFF $B428_H$	0	8, 16, 32
INTC2	EI Level Interrupt Mask Register 11	IMR11	$FFFF FFFF_H$	32	FFFF $B42C_H$	0	8, 16, 32
INTC2	EI Level Interrupt Mask Register 12	IMR12	$FFFF FFFF_H$	32	FFFF $B430_H$	0	8, 16, 32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
INTC2	EI Level Interrupt Mask Register 13	IMR13	FFFF FFFF$_H$	32	FFFF B434$_H$	0	8, 16, 32
INTC2	EI Level Interrupt Mask Register 14	IMR14	FFFF FFFF$_H$	32	FFFF B438$_H$	0	8, 16, 32
INTC2	EI Level Interrupt Mask Register 15	IMR15	FFFF FFFF$_H$	32	FFFF B43C$_H$	0	8, 16, 32
INTC2	EI Level Interrupt Bind Register 33	EIBD33	0000 0001$_H$	32	FFFF B884$_H$	0	32
INTC2	EI Level Interrupt Bind Register 38	EIBD38	0000 0001$_H$	32	FFFF B898$_H$	0	32
INTC2	EI Level Interrupt Bind Register 39	EIBD39	0000 0001$_H$	32	FFFF B89C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 40	EIBD40	0000 0001$_H$	32	FFFF B8A0$_H$	0	32
INTC2	EI Level Interrupt Bind Register 41	EIBD41	0000 0001$_H$	32	FFFF B8A4$_H$	0	32
INTC2	EI Level Interrupt Bind Register 42	EIBD42	0000 0001$_H$	32	FFFF B8A8$_H$	0	32
INTC2	EI Level Interrupt Bind Register 43	EIBD43	0000 0001$_H$	32	FFFF B8AC$_H$	0	32
INTC2	EI Level Interrupt Bind Register 44	EIBD44	0000 0001$_H$	32	FFFF B8B0$_H$	0	32
INTC2	EI Level Interrupt Bind Register 45	EIBD45	0000 0001$_H$	32	FFFF B8B4$_H$	0	32
INTC2	EI Level Interrupt Bind Register 46	EIBD46	0000 0001$_H$	32	FFFF B8B8$_H$	0	32
INTC2	EI Level Interrupt Bind Register 47	EIBD47	0000 0001$_H$	32	FFFF B8BC$_H$	0	32
INTC2	EI Level Interrupt Bind Register 48	EIBD48	0000 0001$_H$	32	FFFF B8C0$_H$	0	32
INTC2	EI Level Interrupt Bind Register 49	EIBD49	0000 0001$_H$	32	FFFF B8C4$_H$	0	32
INTC2	EI Level Interrupt Bind Register 50	EIBD50	0000 0001$_H$	32	FFFF B8C8$_H$	0	32
INTC2	EI Level Interrupt Bind Register 51	EIBD51	0000 0001$_H$	32	FFFF B8CC$_H$	0	32
INTC2	EI Level Interrupt Bind Register 52	EIBD52	0000 0001$_H$	32	FFFF B8D0$_H$	0	32
INTC2	EI Level Interrupt Bind Register 53	EIBD53	0000 0001$_H$	32	FFFF B8D4$_H$	0	32
INTC2	EI Level Interrupt Bind Register 54	EIBD54	0000 0001$_H$	32	FFFF B8D8$_H$	0	32
INTC2	EI Level Interrupt Bind Register 55	EIBD55	0000 0001$_H$	32	FFFF B8DC$_H$	0	32
INTC2	EI Level Interrupt Bind Register 57	EIBD57	0000 0001$_H$	32	FFFF B8E4$_H$	0	32
INTC2	EI Level Interrupt Bind Register 58	EIBD58	0000 0001$_H$	32	FFFF B8E8$_H$	0	32
INTC2	EI Level Interrupt Bind Register 59	EIBD59	0000 0001$_H$	32	FFFF B8EC$_H$	0	32
INTC2	EI Level Interrupt Bind Register 60	EIBD60	0000 0001$_H$	32	FFFF B8F0$_H$	0	32
INTC2	EI Level Interrupt Bind Register 61	EIBD61	0000 0001$_H$	32	FFFF B8F4$_H$	0	32
INTC2	EI Level Interrupt Bind Register 62	EIBD62	0000 0001$_H$	32	FFFF B8F8$_H$	0	32
INTC2	EI Level Interrupt Bind Register 63	EIBD63	0000 0001$_H$	32	FFFF B8FC$_H$	0	32
INTC2	EI Level Interrupt Bind Register 64	EIBD64	0000 0001$_H$	32	FFFF B900$_H$	0	32
INTC2	EI Level Interrupt Bind Register 65	EIBD65	0000 0001$_H$	32	FFFF B904$_H$	0	32
INTC2	EI Level Interrupt Bind Register 66	EIBD66	0000 0001$_H$	32	FFFF B908$_H$	0	32
INTC2	EI Level Interrupt Bind Register 67	EIBD67	0000 0001$_H$	32	FFFF B90C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 68	EIBD68	0000 0001$_H$	32	FFFF B910$_H$	0	32
INTC2	EI Level Interrupt Bind Register 69	EIBD69	0000 0001$_H$	32	FFFF B914$_H$	0	32
INTC2	EI Level Interrupt Bind Register 70	EIBD70	0000 0001$_H$	32	FFFF B918$_H$	0	32
INTC2	EI Level Interrupt Bind Register 71	EIBD71	0000 0001$_H$	32	FFFF B91C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 72	EIBD72	0000 0001$_H$	32	FFFF B920$_H$	0	32
INTC2	EI Level Interrupt Bind Register 73	EIBD73	0000 0001$_H$	32	FFFF B924$_H$	0	32
INTC2	EI Level Interrupt Bind Register 74	EIBD74	0000 0001$_H$	32	FFFF B928$_H$	0	32
INTC2	EI Level Interrupt Bind Register 75	EIBD75	0000 0001$_H$	32	FFFF B92C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 76	EIBD76	0000 0001$_H$	32	FFFF B930$_H$	0	32
INTC2	EI Level Interrupt Bind Register 77	EIBD77	0000 0001$_H$	32	FFFF B934$_H$	0	32
INTC2	EI Level Interrupt Bind Register 78	EIBD78	0000 0001$_H$	32	FFFF B938$_H$	0	32
INTC2	EI Level Interrupt Bind Register 79	EIBD79	0000 0001$_H$	32	FFFF B93C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 80	EIBD80	0000 0001$_H$	32	FFFF B940$_H$	0	32
INTC2	EI Level Interrupt Bind Register 81	EIBD81	0000 0001$_H$	32	FFFF B944$_H$	0	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
INTC2	EI Level Interrupt Bind Register 82	EIBD82	0000 0001$_H$	32	FFFF B948$_H$	0	32
INTC2	EI Level Interrupt Bind Register 83	EIBD83	0000 0001$_H$	32	FFFF B94C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 84	EIBD84	0000 0001$_H$	32	FFFF B950$_H$	0	32
INTC2	EI Level Interrupt Bind Register 85	EIBD85	0000 0001$_H$	32	FFFF B954$_H$	0	32
INTC2	EI Level Interrupt Bind Register 86	EIBD86	0000 0001$_H$	32	FFFF B958$_H$	0	32
INTC2	EI Level Interrupt Bind Register 87	EIBD87	0000 0001$_H$	32	FFFF B95C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 88	EIBD88	0000 0001$_H$	32	FFFF B960$_H$	0	32
INTC2	EI Level Interrupt Bind Register 89	EIBD89	0000 0001$_H$	32	FFFF B964$_H$	0	32
INTC2	EI Level Interrupt Bind Register 90	EIBD90	0000 0001$_H$	32	FFFF B968$_H$	0	32
INTC2	EI Level Interrupt Bind Register 91	EIBD91	0000 0001$_H$	32	FFFF B96C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 92	EIBD92	0000 0001$_H$	32	FFFF B970$_H$	0	32
INTC2	EI Level Interrupt Bind Register 93	EIBD93	0000 0001$_H$	32	FFFF B974$_H$	0	32
INTC2	EI Level Interrupt Bind Register 94	EIBD94	0000 0001$_H$	32	FFFF B978$_H$	0	32
INTC2	EI Level Interrupt Bind Register 95	EIBD95	0000 0001$_H$	32	FFFF B97C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 96	EIBD96	0000 0001$_H$	32	FFFF B980$_H$	0	32
INTC2	EI Level Interrupt Bind Register 97	EIBD97	0000 0001$_H$	32	FFFF B984$_H$	0	32
INTC2	EI Level Interrupt Bind Register 98	EIBD98	0000 0001$_H$	32	FFFF B988$_H$	0	32
INTC2	EI Level Interrupt Bind Register 99	EIBD99	0000 0001$_H$	32	FFFF B98C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 100	EIBD100	0000 0001$_H$	32	FFFF B990$_H$	0	32
INTC2	EI Level Interrupt Bind Register 101	EIBD101	0000 0001$_H$	32	FFFF B994$_H$	0	32
INTC2	EI Level Interrupt Bind Register 102	EIBD102	0000 0001$_H$	32	FFFF B998$_H$	0	32
INTC2	EI Level Interrupt Bind Register 103	EIBD103	0000 0001$_H$	32	FFFF B99C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 104	EIBD104	0000 0001$_H$	32	FFFF B9A0$_H$	0	32
INTC2	EI Level Interrupt Bind Register 105	EIBD105	0000 0001$_H$	32	FFFF B9A4$_H$	0	32
INTC2	EI Level Interrupt Bind Register 106	EIBD106	0000 0001$_H$	32	FFFF B9A8$_H$	0	32
INTC2	EI Level Interrupt Bind Register 107	EIBD107	0000 0001$_H$	32	FFFF B9AC$_H$	0	32
INTC2	EI Level Interrupt Bind Register 108	EIBD108	0000 0001$_H$	32	FFFF B9B0$_H$	0	32
INTC2	EI Level Interrupt Bind Register 109	EIBD109	0000 0001$_H$	32	FFFF B9B4$_H$	0	32
INTC2	EI Level Interrupt Bind Register 110	EIBD110	0000 0001$_H$	32	FFFF B9B8$_H$	0	32
INTC2	EI Level Interrupt Bind Register 111	EIBD111	0000 0001$_H$	32	FFFF B9BC$_H$	0	32
INTC2	EI Level Interrupt Bind Register 112	EIBD112	0000 0001$_H$	32	FFFF B9C0$_H$	0	32
INTC2	EI Level Interrupt Bind Register 113	EIBD113	0000 0001$_H$	32	FFFF B9C4$_H$	0	32
INTC2	EI Level Interrupt Bind Register 114	EIBD114	0000 0001$_H$	32	FFFF B9C8$_H$	0	32
INTC2	EI Level Interrupt Bind Register 115	EIBD115	0000 0001$_H$	32	FFFF B9CC$_H$	0	32
INTC2	EI Level Interrupt Bind Register 116	EIBD116	0000 0001$_H$	32	FFFF B9D0$_H$	0	32
INTC2	EI Level Interrupt Bind Register 117	EIBD117	0000 0001$_H$	32	FFFF B9D4$_H$	0	32
INTC2	EI Level Interrupt Bind Register 118	EIBD118	0000 0001$_H$	32	FFFF B9D8$_H$	0	32
INTC2	EI Level Interrupt Bind Register 119	EIBD119	0000 0001$_H$	32	FFFF B9DC$_H$	0	32
INTC2	EI Level Interrupt Bind Register 120	EIBD120	0000 0001$_H$	32	FFFF B9E0$_H$	0	32
INTC2	EI Level Interrupt Bind Register 121	EIBD121	0000 0001$_H$	32	FFFF B9E4$_H$	0	32
INTC2	EI Level Interrupt Bind Register 122	EIBD122	0000 0001$_H$	32	FFFF B9E8$_H$	0	32
INTC2	EI Level Interrupt Bind Register 123	EIBD123	0000 0001$_H$	32	FFFF B9EC$_H$	0	32
INTC2	EI Level Interrupt Bind Register 124	EIBD124	0000 0001$_H$	32	FFFF B9F0$_H$	0	32
INTC2	EI Level Interrupt Bind Register 125	EIBD125	0000 0001$_H$	32	FFFF B9F4$_H$	0	32
INTC2	EI Level Interrupt Bind Register 126	EIBD126	0000 0001$_H$	32	FFFF B9F8$_H$	0	32
INTC2	EI Level Interrupt Bind Register 127	EIBD127	0000 0001$_H$	32	FFFF B9FC$_H$	0	32
INTC2	EI Level Interrupt Bind Register 128	EIBD128	0000 0001$_H$	32	FFFF BA00$_H$	0	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
INTC2	EI Level Interrupt Bind Register 129	EIBD129	0000 0001$_H$	32	FFFF BA04$_H$	0	32
INTC2	EI Level Interrupt Bind Register 130	EIBD130	0000 0001$_H$	32	FFFF BA08$_H$	0	32
INTC2	EI Level Interrupt Bind Register 131	EIBD131	0000 0001$_H$	32	FFFF BA0C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 132	EIBD132	0000 0001$_H$	32	FFFF BA10$_H$	0	32
INTC2	EI Level Interrupt Bind Register 133	EIBD133	0000 0001$_H$	32	FFFF BA14$_H$	0	32
INTC2	EI Level Interrupt Bind Register 134	EIBD134	0000 0001$_H$	32	FFFF BA18$_H$	0	32
INTC2	EI Level Interrupt Bind Register 135	EIBD135	0000 0001$_H$	32	FFFF BA1C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 136	EIBD136	0000 0001$_H$	32	FFFF BA20$_H$	0	32
INTC2	EI Level Interrupt Bind Register 137	EIBD137	0000 0001$_H$	32	FFFF BA24$_H$	0	32
INTC2	EI Level Interrupt Bind Register 138	EIBD138	0000 0001$_H$	32	FFFF BA28$_H$	0	32
INTC2	EI Level Interrupt Bind Register 139	EIBD139	0000 0001$_H$	32	FFFF BA2C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 140	EIBD140	0000 0001$_H$	32	FFFF BA30$_H$	0	32
INTC2	EI Level Interrupt Bind Register 141	EIBD141	0000 0001$_H$	32	FFFF BA34$_H$	0	32
INTC2	EI Level Interrupt Bind Register 142	EIBD142	0000 0001$_H$	32	FFFF BA38$_H$	0	32
INTC2	EI Level Interrupt Bind Register 143	EIBD143	0000 0001$_H$	32	FFFF BA3C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 144	EIBD144	0000 0001$_H$	32	FFFF BA40$_H$	0	32
INTC2	EI Level Interrupt Bind Register 145	EIBD145	0000 0001$_H$	32	FFFF BA44$_H$	0	32
INTC2	EI Level Interrupt Bind Register 146	EIBD146	0000 0001$_H$	32	FFFF BA48$_H$	0	32
INTC2	EI Level Interrupt Bind Register 147	EIBD147	0000 0001$_H$	32	FFFF BA4C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 148	EIBD148	0000 0001$_H$	32	FFFF BA50$_H$	0	32
INTC2	EI Level Interrupt Bind Register 149	EIBD149	0000 0001$_H$	32	FFFF BA54$_H$	0	32
INTC2	EI Level Interrupt Bind Register 150	EIBD150	0000 0001$_H$	32	FFFF BA58$_H$	0	32
INTC2	EI Level Interrupt Bind Register 151	EIBD151	0000 0001$_H$	32	FFFF BA5C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 152	EIBD152	0000 0001$_H$	32	FFFF BA60$_H$	0	32
INTC2	EI Level Interrupt Bind Register 153	EIBD153	0000 0001$_H$	32	FFFF BA64$_H$	0	32
INTC2	EI Level Interrupt Bind Register 154	EIBD154	0000 0001$_H$	32	FFFF BA68$_H$	0	32
INTC2	EI Level Interrupt Bind Register 155	EIBD155	0000 0001$_H$	32	FFFF BA6C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 156	EIBD156	0000 0001$_H$	32	FFFF BA70$_H$	0	32
INTC2	EI Level Interrupt Bind Register 157	EIBD157	0000 0001$_H$	32	FFFF BA74$_H$	0	32
INTC2	EI Level Interrupt Bind Register 158	EIBD158	0000 0001$_H$	32	FFFF BA78$_H$	0	32
INTC2	EI Level Interrupt Bind Register 159	EIBD159	0000 0001$_H$	32	FFFF BA7C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 160	EIBD160	0000 0001$_H$	32	FFFF BA80$_H$	0	32
INTC2	EI Level Interrupt Bind Register 161	EIBD161	0000 0001$_H$	32	FFFF BA84$_H$	0	32
INTC2	EI Level Interrupt Bind Register 162	EIBD162	0000 0001$_H$	32	FFFF BA88$_H$	0	32
INTC2	EI Level Interrupt Bind Register 163	EIBD163	0000 0001$_H$	32	FFFF BA8C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 164	EIBD164	0000 0001$_H$	32	FFFF BA90$_H$	0	32
INTC2	EI Level Interrupt Bind Register 165	EIBD165	0000 0001$_H$	32	FFFF BA94$_H$	0	32
INTC2	EI Level Interrupt Bind Register 166	EIBD166	0000 0001$_H$	32	FFFF BA98$_H$	0	32
INTC2	EI Level Interrupt Bind Register 167	EIBD167	0000 0001$_H$	32	FFFF BA9C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 168	EIBD168	0000 0001$_H$	32	FFFF BAA0$_H$	0	32
INTC2	EI Level Interrupt Bind Register 169	EIBD169	0000 0001$_H$	32	FFFF BAA4$_H$	0	32
INTC2	EI Level Interrupt Bind Register 170	EIBD170	0000 0001$_H$	32	FFFF BAA8$_H$	0	32
INTC2	EI Level Interrupt Bind Register 171	EIBD171	0000 0001$_H$	32	FFFF BAAC$_H$	0	32
INTC2	EI Level Interrupt Bind Register 172	EIBD172	0000 0001$_H$	32	FFFF BAB0$_H$	0	32
INTC2	EI Level Interrupt Bind Register 173	EIBD173	0000 0001$_H$	32	FFFF BAB4$_H$	0	32
INTC2	EI Level Interrupt Bind Register 174	EIBD174	0000 0001$_H$	32	FFFF BAB8$_H$	0	32
INTC2	EI Level Interrupt Bind Register 175	EIBD175	0000 0001$_H$	32	FFFF BABC$_H$	0	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
INTC2	EI Level Interrupt Bind Register 176	EIBD176	0000 0001$_H$	32	FFFF BAC0$_H$	0	32
INTC2	EI Level Interrupt Bind Register 177	EIBD177	0000 0001$_H$	32	FFFF BAC4$_H$	0	32
INTC2	EI Level Interrupt Bind Register 178	EIBD178	0000 0001$_H$	32	FFFF BAC8$_H$	0	32
INTC2	EI Level Interrupt Bind Register 179	EIBD179	0000 0001$_H$	32	FFFF BACC$_H$	0	32
INTC2	EI Level Interrupt Bind Register 180	EIBD180	0000 0001$_H$	32	FFFF BAD0$_H$	0	32
INTC2	EI Level Interrupt Bind Register 181	EIBD181	0000 0001$_H$	32	FFFF BAD4$_H$	0	32
INTC2	EI Level Interrupt Bind Register 182	EIBD182	0000 0001$_H$	32	FFFF BAD8$_H$	0	32
INTC2	EI Level Interrupt Bind Register 183	EIBD183	0000 0001$_H$	32	FFFF BADC$_H$	0	32
INTC2	EI Level Interrupt Bind Register 184	EIBD184	0000 0001$_H$	32	FFFF BAE0$_H$	0	32
INTC2	EI Level Interrupt Bind Register 185	EIBD185	0000 0001$_H$	32	FFFF BAE4$_H$	0	32
INTC2	EI Level Interrupt Bind Register 186	EIBD186	0000 0001$_H$	32	FFFF BAE8$_H$	0	32
INTC2	EI Level Interrupt Bind Register 187	EIBD187	0000 0001$_H$	32	FFFF BAEC$_H$	0	32
INTC2	EI Level Interrupt Bind Register 188	EIBD188	0000 0001$_H$	32	FFFF BAF0$_H$	0	32
INTC2	EI Level Interrupt Bind Register 189	EIBD189	0000 0001$_H$	32	FFFF BAF4$_H$	0	32
INTC2	EI Level Interrupt Bind Register 190	EIBD190	0000 0001$_H$	32	FFFF BAF8$_H$	0	32
INTC2	EI Level Interrupt Bind Register 191	EIBD191	0000 0001$_H$	32	FFFF BAFC$_H$	0	32
INTC2	EI Level Interrupt Bind Register 192	EIBD192	0000 0001$_H$	32	FFFF BB00$_H$	0	32
INTC2	EI Level Interrupt Bind Register 193	EIBD193	0000 0001$_H$	32	FFFF BB04$_H$	0	32
INTC2	EI Level Interrupt Bind Register 194	EIBD194	0000 0001$_H$	32	FFFF BB08$_H$	0	32
INTC2	EI Level Interrupt Bind Register 195	EIBD195	0000 0001$_H$	32	FFFF BB0C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 196	EIBD196	0000 0001$_H$	32	FFFF BB10$_H$	0	32
INTC2	EI Level Interrupt Bind Register 197	EIBD197	0000 0001$_H$	32	FFFF BB14$_H$	0	32
INTC2	EI Level Interrupt Bind Register 198	EIBD198	0000 0001$_H$	32	FFFF BB18$_H$	0	32
INTC2	EI Level Interrupt Bind Register 199	EIBD199	0000 0001$_H$	32	FFFF BB1C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 200	EIBD200	0000 0001$_H$	32	FFFF BB20$_H$	0	32
INTC2	EI Level Interrupt Bind Register 201	EIBD201	0000 0001$_H$	32	FFFF BB24$_H$	0	32
INTC2	EI Level Interrupt Bind Register 202	EIBD202	0000 0001$_H$	32	FFFF BB28$_H$	0	32
INTC2	EI Level Interrupt Bind Register 203	EIBD203	0000 0001$_H$	32	FFFF BB2C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 204	EIBD204	0000 0001$_H$	32	FFFF BB30$_H$	0	32
INTC2	EI Level Interrupt Bind Register 205	EIBD205	0000 0001$_H$	32	FFFF BB34$_H$	0	32
INTC2	EI Level Interrupt Bind Register 206	EIBD206	0000 0001$_H$	32	FFFF BB38$_H$	0	32
INTC2	EI Level Interrupt Bind Register 207	EIBD207	0000 0001$_H$	32	FFFF BB3C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 208	EIBD208	0000 0001$_H$	32	FFFF BB40$_H$	0	32
INTC2	EI Level Interrupt Bind Register 209	EIBD209	0000 0001$_H$	32	FFFF BB44$_H$	0	32
INTC2	EI Level Interrupt Bind Register 210	EIBD210	0000 0001$_H$	32	FFFF BB48$_H$	0	32
INTC2	EI Level Interrupt Bind Register 211	EIBD211	0000 0001$_H$	32	FFFF BB4C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 212	EIBD212	0000 0001$_H$	32	FFFF BB50$_H$	0	32
INTC2	EI Level Interrupt Bind Register 213	EIBD213	0000 0001$_H$	32	FFFF BB54$_H$	0	32
INTC2	EI Level Interrupt Bind Register 214	EIBD214	0000 0001$_H$	32	FFFF BB58$_H$	0	32
INTC2	EI Level Interrupt Bind Register 215	EIBD215	0000 0001$_H$	32	FFFF BB5C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 216	EIBD216	0000 0001$_H$	32	FFFF BB60$_H$	0	32
INTC2	EI Level Interrupt Bind Register 217	EIBD217	0000 0001$_H$	32	FFFF BB64$_H$	0	32
INTC2	EI Level Interrupt Bind Register 218	EIBD218	0000 0001$_H$	32	FFFF BB68$_H$	0	32
INTC2	EI Level Interrupt Bind Register 219	EIBD219	0000 0001$_H$	32	FFFF BB6C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 220	EIBD220	0000 0001$_H$	32	FFFF BB70$_H$	0	32
INTC2	EI Level Interrupt Bind Register 221	EIBD221	0000 0001$_H$	32	FFFF BB74$_H$	0	32
INTC2	EI Level Interrupt Bind Register 222	EIBD222	0000 0001$_H$	32	FFFF BB78$_H$	0	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
INTC2	EI Level Interrupt Bind Register 223	EIBD223	0000 0001$_H$	32	FFFF BB7C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 224	EIBD224	0000 0001$_H$	32	FFFF BB80$_H$	0	32
INTC2	EI Level Interrupt Bind Register 225	EIBD225	0000 0001$_H$	32	FFFF BB84$_H$	0	32
INTC2	EI Level Interrupt Bind Register 226	EIBD226	0000 0001$_H$	32	FFFF BB88$_H$	0	32
INTC2	EI Level Interrupt Bind Register 227	EIBD227	0000 0001$_H$	32	FFFF BB8C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 228	EIBD228	0000 0001$_H$	32	FFFF BB90$_H$	0	32
INTC2	EI Level Interrupt Bind Register 229	EIBD229	0000 0001$_H$	32	FFFF BB94$_H$	0	32
INTC2	EI Level Interrupt Bind Register 230	EIBD230	0000 0001$_H$	32	FFFF BB98$_H$	0	32
INTC2	EI Level Interrupt Bind Register 231	EIBD231	0000 0001$_H$	32	FFFF BB9C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 232	EIBD232	0000 0001$_H$	32	FFFF BBA0$_H$	0	32
INTC2	EI Level Interrupt Bind Register 233	EIBD233	0000 0001$_H$	32	FFFF BBA4$_H$	0	32
INTC2	EI Level Interrupt Bind Register 234	EIBD234	0000 0001$_H$	32	FFFF BBA8$_H$	0	32
INTC2	EI Level Interrupt Bind Register 235	EIBD235	0000 0001$_H$	32	FFFF BBAC$_H$	0	32
INTC2	EI Level Interrupt Bind Register 236	EIBD236	0000 0001$_H$	32	FFFF BBB0$_H$	0	32
INTC2	EI Level Interrupt Bind Register 237	EIBD237	0000 0001$_H$	32	FFFF BBB4$_H$	0	32
INTC2	EI Level Interrupt Bind Register 238	EIBD238	0000 0001$_H$	32	FFFF BBB8$_H$	0	32
INTC2	EI Level Interrupt Bind Register 239	EIBD239	0000 0001$_H$	32	FFFF BBBC$_H$	0	32
INTC2	EI Level Interrupt Bind Register 240	EIBD240	0000 0001$_H$	32	FFFF BBC0$_H$	0	32
INTC2	EI Level Interrupt Bind Register 241	EIBD241	0000 0001$_H$	32	FFFF BBC4$_H$	0	32
INTC2	EI Level Interrupt Bind Register 242	EIBD242	0000 0001$_H$	32	FFFF BBC8$_H$	0	32
INTC2	EI Level Interrupt Bind Register 243	EIBD243	0000 0001$_H$	32	FFFF BBCC$_H$	0	32
INTC2	EI Level Interrupt Bind Register 244	EIBD244	0000 0001$_H$	32	FFFF BBD0$_H$	0	32
INTC2	EI Level Interrupt Bind Register 245	EIBD245	0000 0001$_H$	32	FFFF BBD4$_H$	0	32
INTC2	EI Level Interrupt Bind Register 246	EIBD246	0000 0001$_H$	32	FFFF BBD8$_H$	0	32
INTC2	EI Level Interrupt Bind Register 247	EIBD247	0000 0001$_H$	32	FFFF BBDC$_H$	0	32
INTC2	EI Level Interrupt Bind Register 248	EIBD248	0000 0001$_H$	32	FFFF BBE0$_H$	0	32
INTC2	EI Level Interrupt Bind Register 249	EIBD249	0000 0001$_H$	32	FFFF BBE4$_H$	0	32
INTC2	EI Level Interrupt Bind Register 250	EIBD250	0000 0001$_H$	32	FFFF BBE8$_H$	0	32
INTC2	EI Level Interrupt Bind Register 251	EIBD251	0000 0001$_H$	32	FFFF BBEC$_H$	0	32
INTC2	EI Level Interrupt Bind Register 252	EIBD252	0000 0001$_H$	32	FFFF BBF0$_H$	0	32
INTC2	EI Level Interrupt Bind Register 253	EIBD253	0000 0001$_H$	32	FFFF BBF4$_H$	0	32
INTC2	EI Level Interrupt Bind Register 254	EIBD254	0000 0001$_H$	32	FFFF BBF8$_H$	0	32
INTC2	EI Level Interrupt Bind Register 255	EIBD255	0000 0001$_H$	32	FFFF BBFC$_H$	0	32
INTC2	EI Level Interrupt Bind Register 256	EIBD256	0000 0001$_H$	32	FFFF BC00$_H$	0	32
INTC2	EI Level Interrupt Bind Register 257	EIBD257	0000 0001$_H$	32	FFFF BC04$_H$	0	32
INTC2	EI Level Interrupt Bind Register 258	EIBD258	0000 0001$_H$	32	FFFF BC08$_H$	0	32
INTC2	EI Level Interrupt Bind Register 259	EIBD259	0000 0001$_H$	32	FFFF BC0C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 260	EIBD260	0000 0001$_H$	32	FFFF BC10$_H$	0	32
INTC2	EI Level Interrupt Bind Register 261	EIBD261	0000 0001$_H$	32	FFFF BC14$_H$	0	32
INTC2	EI Level Interrupt Bind Register 262	EIBD262	0000 0001$_H$	32	FFFF BC18$_H$	0	32
INTC2	EI Level Interrupt Bind Register 263	EIBD263	0000 0001$_H$	32	FFFF BC1C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 264	EIBD264	0000 0001$_H$	32	FFFF BC20$_H$	0	32
INTC2	EI Level Interrupt Bind Register 265	EIBD265	0000 0001$_H$	32	FFFF BC24$_H$	0	32
INTC2	EI Level Interrupt Bind Register 266	EIBD266	0000 0001$_H$	32	FFFF BC28$_H$	0	32
INTC2	EI Level Interrupt Bind Register 267	EIBD267	0000 0001$_H$	32	FFFF BC2C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 268	EIBD268	0000 0001$_H$	32	FFFF BC30$_H$	0	32
INTC2	EI Level Interrupt Bind Register 269	EIBD269	0000 0001$_H$	32	FFFF BC34$_H$	0	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
INTC2	EI Level Interrupt Bind Register 270	EIBD270	0000 0001$_H$	32	FFFF BC38$_H$	0	32
INTC2	EI Level Interrupt Bind Register 271	EIBD271	0000 0001$_H$	32	FFFF BC3C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 272	EIBD272	0000 0001$_H$	32	FFFF BC40$_H$	0	32
INTC2	EI Level Interrupt Bind Register 273	EIBD273	0000 0001$_H$	32	FFFF BC44$_H$	0	32
INTC2	EI Level Interrupt Bind Register 274	EIBD274	0000 0001$_H$	32	FFFF BC48$_H$	0	32
INTC2	EI Level Interrupt Bind Register 275	EIBD275	0000 0001$_H$	32	FFFF BC4C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 276	EIBD276	0000 0001$_H$	32	FFFF BC50$_H$	0	32
INTC2	EI Level Interrupt Bind Register 277	EIBD277	0000 0001$_H$	32	FFFF BC54$_H$	0	32
INTC2	EI Level Interrupt Bind Register 278	EIBD278	0000 0001$_H$	32	FFFF BC58$_H$	0	32
INTC2	EI Level Interrupt Bind Register 279	EIBD279	0000 0001$_H$	32	FFFF BC5C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 280	EIBD280	0000 0001$_H$	32	FFFF BC60$_H$	0	32
INTC2	EI Level Interrupt Bind Register 281	EIBD281	0000 0001$_H$	32	FFFF BC64$_H$	0	32
INTC2	EI Level Interrupt Bind Register 282	EIBD282	0000 0001$_H$	32	FFFF BC68$_H$	0	32
INTC2	EI Level Interrupt Bind Register 283	EIBD283	0000 0001$_H$	32	FFFF BC6C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 284	EIBD284	0000 0001$_H$	32	FFFF BC70$_H$	0	32
INTC2	EI Level Interrupt Bind Register 285	EIBD285	0000 0001$_H$	32	FFFF BC74$_H$	0	32
INTC2	EI Level Interrupt Bind Register 286	EIBD286	0000 0001$_H$	32	FFFF BC78$_H$	0	32
INTC2	EI Level Interrupt Bind Register 287	EIBD287	0000 0001$_H$	32	FFFF BC7C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 288	EIBD288	0000 0001$_H$	32	FFFF BC80$_H$	0	32
INTC2	EI Level Interrupt Bind Register 289	EIBD289	0000 0001$_H$	32	FFFF BC84$_H$	0	32
INTC2	EI Level Interrupt Bind Register 290	EIBD290	0000 0001$_H$	32	FFFF BC88$_H$	0	32
INTC2	EI Level Interrupt Bind Register 291	EIBD291	0000 0001$_H$	32	FFFF BC8C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 292	EIBD292	0000 0001$_H$	32	FFFF BC90$_H$	0	32
INTC2	EI Level Interrupt Bind Register 293	EIBD293	0000 0001$_H$	32	FFFF BC94$_H$	0	32
INTC2	EI Level Interrupt Bind Register 294	EIBD294	0000 0001$_H$	32	FFFF BC98$_H$	0	32
INTC2	EI Level Interrupt Bind Register 295	EIBD295	0000 0001$_H$	32	FFFF BC9C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 296	EIBD296	0000 0001$_H$	32	FFFF BCA0$_H$	0	32
INTC2	EI Level Interrupt Bind Register 297	EIBD297	0000 0001$_H$	32	FFFF BCA4$_H$	0	32
INTC2	EI Level Interrupt Bind Register 298	EIBD298	0000 0001$_H$	32	FFFF BCA8$_H$	0	32
INTC2	EI Level Interrupt Bind Register 299	EIBD299	0000 0001$_H$	32	FFFF BCAC$_H$	0	32
INTC2	EI Level Interrupt Bind Register 300	EIBD300	0000 0001$_H$	32	FFFF BCB0$_H$	0	32
INTC2	EI Level Interrupt Bind Register 301	EIBD301	0000 0001$_H$	32	FFFF BCB4$_H$	0	32
INTC2	EI Level Interrupt Bind Register 302	EIBD302	0000 0001$_H$	32	FFFF BCB8$_H$	0	32
INTC2	EI Level Interrupt Bind Register 303	EIBD303	0000 0001$_H$	32	FFFF BCBC$_H$	0	32
INTC2	EI Level Interrupt Bind Register 304	EIBD304	0000 0001$_H$	32	FFFF BCC0$_H$	0	32
INTC2	EI Level Interrupt Bind Register 305	EIBD305	0000 0001$_H$	32	FFFF BCC4$_H$	0	32
INTC2	EI Level Interrupt Bind Register 306	EIBD306	0000 0001$_H$	32	FFFF BCC8$_H$	0	32
INTC2	EI Level Interrupt Bind Register 307	EIBD307	0000 0001$_H$	32	FFFF BCCC$_H$	0	32
INTC2	EI Level Interrupt Bind Register 308	EIBD308	0000 0001$_H$	32	FFFF BCD0$_H$	0	32
INTC2	EI Level Interrupt Bind Register 309	EIBD309	0000 0001$_H$	32	FFFF BCD4$_H$	0	32
INTC2	EI Level Interrupt Bind Register 310	EIBD310	0000 0001$_H$	32	FFFF BCD8$_H$	0	32
INTC2	EI Level Interrupt Bind Register 311	EIBD311	0000 0001$_H$	32	FFFF BCDC$_H$	0	32
INTC2	EI Level Interrupt Bind Register 312	EIBD312	0000 0001$_H$	32	FFFF BCE0$_H$	0	32
INTC2	EI Level Interrupt Bind Register 313	EIBD313	0000 0001$_H$	32	FFFF BCE4$_H$	0	32
INTC2	EI Level Interrupt Bind Register 314	EIBD314	0000 0001$_H$	32	FFFF BCE8$_H$	0	32
INTC2	EI Level Interrupt Bind Register 315	EIBD315	0000 0001$_H$	32	FFFF BCEC$_H$	0	32
INTC2	EI Level Interrupt Bind Register 316	EIBD316	0000 0001$_H$	32	FFFF BCF0$_H$	0	32

(198/201)

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
INTC2	EI Level Interrupt Bind Register 317	EIBD317	0000 0001$_H$	32	FFFF BCF4$_H$	0	32
INTC2	EI Level Interrupt Bind Register 318	EIBD318	0000 0001$_H$	32	FFFF BCF8$_H$	0	32
INTC2	EI Level Interrupt Bind Register 321	EIBD321	0000 0001$_H$	32	FFFF BD04$_H$	0	32
INTC2	EI Level Interrupt Bind Register 322	EIBD322	0000 0001$_H$	32	FFFF BD08$_H$	0	32
INTC2	EI Level Interrupt Bind Register 323	EIBD323	0000 0001$_H$	32	FFFF BD0C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 324	EIBD324	0000 0001$_H$	32	FFFF BD10$_H$	0	32
INTC2	EI Level Interrupt Bind Register 325	EIBD325	0000 0001$_H$	32	FFFF BD14$_H$	0	32
INTC2	EI Level Interrupt Bind Register 326	EIBD326	0000 0001$_H$	32	FFFF BD18$_H$	0	32
INTC2	EI Level Interrupt Bind Register 329	EIBD329	0000 0001$_H$	32	FFFF BD24$_H$	0	32
INTC2	EI Level Interrupt Bind Register 330	EIBD330	0000 0001$_H$	32	FFFF BD28$_H$	0	32
INTC2	EI Level Interrupt Bind Register 331	EIBD331	0000 0001$_H$	32	FFFF BD2C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 332	EIBD332	0000 0001$_H$	32	FFFF BD30$_H$	0	32
INTC2	EI Level Interrupt Bind Register 333	EIBD333	0000 0001$_H$	32	FFFF BD34$_H$	0	32
INTC2	EI Level Interrupt Bind Register 334	EIBD334	0000 0001$_H$	32	FFFF BD38$_H$	0	32
INTC2	EI Level Interrupt Bind Register 337	EIBD337	0000 0001$_H$	32	FFFF BD44$_H$	0	32
INTC2	EI Level Interrupt Bind Register 338	EIBD338	0000 0001$_H$	32	FFFF BD48$_H$	0	32
INTC2	EI Level Interrupt Bind Register 339	EIBD339	0000 0001$_H$	32	FFFF BD4C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 340	EIBD340	0000 0001$_H$	32	FFFF BD50$_H$	0	32
INTC2	EI Level Interrupt Bind Register 341	EIBD341	0000 0001$_H$	32	FFFF BD54$_H$	0	32
INTC2	EI Level Interrupt Bind Register 342	EIBD342	0000 0001$_H$	32	FFFF BD58$_H$	0	32
INTC2	EI Level Interrupt Bind Register 343	EIBD343	0000 0001$_H$	32	FFFF BD5C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 344	EIBD344	0000 0001$_H$	32	FFFF BD60$_H$	0	32
INTC2	EI Level Interrupt Bind Register 345	EIBD345	0000 0001$_H$	32	FFFF BD64$_H$	0	32
INTC2	EI Level Interrupt Bind Register 346	EIBD346	0000 0001$_H$	32	FFFF BD68$_H$	0	32
INTC2	EI Level Interrupt Bind Register 347	EIBD347	0000 0001$_H$	32	FFFF BD6C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 348	EIBD348	0000 0001$_H$	32	FFFF BD70$_H$	0	32
INTC2	EI Level Interrupt Bind Register 349	EIBD349	0000 0001$_H$	32	FFFF BD74$_H$	0	32
INTC2	EI Level Interrupt Bind Register 350	EIBD350	0000 0001$_H$	32	FFFF BD78$_H$	0	32
INTC2	EI Level Interrupt Bind Register 351	EIBD351	0000 0001$_H$	32	FFFF BD7C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 352	EIBD352	0000 0001$_H$	32	FFFF BD80$_H$	0	32
INTC2	EI Level Interrupt Bind Register 353	EIBD353	0000 0001$_H$	32	FFFF BD84$_H$	0	32
INTC2	EI Level Interrupt Bind Register 354	EIBD354	0000 0001$_H$	32	FFFF BD88$_H$	0	32
INTC2	EI Level Interrupt Bind Register 355	EIBD355	0000 0001$_H$	32	FFFF BD8C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 356	EIBD356	0000 0001$_H$	32	FFFF BD90$_H$	0	32
INTC2	EI Level Interrupt Bind Register 357	EIBD357	0000 0001$_H$	32	FFFF BD94$_H$	0	32
INTC2	EI Level Interrupt Bind Register 358	EIBD358	0000 0001$_H$	32	FFFF BD98$_H$	0	32
INTC2	EI Level Interrupt Bind Register 359	EIBD359	0000 0001$_H$	32	FFFF BD9C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 360	EIBD360	0000 0001$_H$	32	FFFF BDA0$_H$	0	32
INTC2	EI Level Interrupt Bind Register 361	EIBD361	0000 0001$_H$	32	FFFF BDA4$_H$	0	32
INTC2	EI Level Interrupt Bind Register 362	EIBD362	0000 0001$_H$	32	FFFF BDA8$_H$	0	32
INTC2	EI Level Interrupt Bind Register 363	EIBD363	0000 0001$_H$	32	FFFF BDAC$_H$	0	32
INTC2	EI Level Interrupt Bind Register 364	EIBD364	0000 0001$_H$	32	FFFF BDB0$_H$	0	32
INTC2	EI Level Interrupt Bind Register 365	EIBD365	0000 0001$_H$	32	FFFF BDB4$_H$	0	32
INTC2	EI Level Interrupt Bind Register 366	EIBD366	0000 0001$_H$	32	FFFF BDB8$_H$	0	32
INTC2	EI Level Interrupt Bind Register 367	EIBD367	0000 0001$_H$	32	FFFF BDBC$_H$	0	32
INTC2	EI Level Interrupt Bind Register 368	EIBD368	0000 0001$_H$	32	FFFF BDC0$_H$	0	32
INTC2	EI Level Interrupt Bind Register 369	EIBD369	0000 0001$_H$	32	FFFF BDC4$_H$	0	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
INTC2	EI Level Interrupt Bind Register 370	EIBD370	0000 0001$_H$	32	FFFF BDC8$_H$	0	32
INTC2	EI Level Interrupt Bind Register 371	EIBD371	0000 0001$_H$	32	FFFF BDCC$_H$	0	32
INTC2	EI Level Interrupt Bind Register 372	EIBD372	0000 0001$_H$	32	FFFF BDD0$_H$	0	32
INTC2	EI Level Interrupt Bind Register 373	EIBD373	0000 0001$_H$	32	FFFF BDD4$_H$	0	32
INTC2	EI Level Interrupt Bind Register 374	EIBD374	0000 0001$_H$	32	FFFF BDD8$_H$	0	32
INTC2	EI Level Interrupt Bind Register 375	EIBD375	0000 0001$_H$	32	FFFF BDDC$_H$	0	32
INTC2	EI Level Interrupt Bind Register 376	EIBD376	0000 0001$_H$	32	FFFF BDE0$_H$	0	32
INTC2	EI Level Interrupt Bind Register 377	EIBD377	0000 0001$_H$	32	FFFF BDE4$_H$	0	32
INTC2	EI Level Interrupt Bind Register 378	EIBD378	0000 0001$_H$	32	FFFF BDE8$_H$	0	32
INTC2	EI Level Interrupt Bind Register 379	EIBD379	0000 0001$_H$	32	FFFF BDEC$_H$	0	32
INTC2	EI Level Interrupt Bind Register 380	EIBD380	0000 0001$_H$	32	FFFF BDF0$_H$	0	32
INTC2	EI Level Interrupt Bind Register 381	EIBD381	0000 0001$_H$	32	FFFF BDF4$_H$	0	32
INTC2	EI Level Interrupt Bind Register 382	EIBD382	0000 0001$_H$	32	FFFF BDF8$_H$	0	32
INTC2	EI Level Interrupt Bind Register 383	EIBD383	0000 0001$_H$	32	FFFF BDFC$_H$	0	32
INTC2	EI Level Interrupt Bind Register 384	EIBD384	0000 0001$_H$	32	FFFF BE00$_H$	0	32
INTC2	EI Level Interrupt Bind Register 385	EIBD385	0000 0001$_H$	32	FFFF BE04$_H$	0	32
INTC2	EI Level Interrupt Bind Register 386	EIBD386	0000 0001$_H$	32	FFFF BE08$_H$	0	32
INTC2	EI Level Interrupt Bind Register 387	EIBD387	0000 0001$_H$	32	FFFF BE0C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 388	EIBD388	0000 0001$_H$	32	FFFF BE10$_H$	0	32
INTC2	EI Level Interrupt Bind Register 389	EIBD389	0000 0001$_H$	32	FFFF BE14$_H$	0	32
INTC2	EI Level Interrupt Bind Register 390	EIBD390	0000 0001$_H$	32	FFFF BE18$_H$	0	32
INTC2	EI Level Interrupt Bind Register 391	EIBD391	0000 0001$_H$	32	FFFF BE1C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 392	EIBD392	0000 0001$_H$	32	FFFF BE20$_H$	0	32
INTC2	EI Level Interrupt Bind Register 393	EIBD393	0000 0001$_H$	32	FFFF BE24$_H$	0	32
INTC2	EI Level Interrupt Bind Register 394	EIBD394	0000 0001$_H$	32	FFFF BE28$_H$	0	32
INTC2	EI Level Interrupt Bind Register 395	EIBD395	0000 0001$_H$	32	FFFF BE2C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 396	EIBD396	0000 0001$_H$	32	FFFF BE30$_H$	0	32
INTC2	EI Level Interrupt Bind Register 397	EIBD397	0000 0001$_H$	32	FFFF BE34$_H$	0	32
INTC2	EI Level Interrupt Bind Register 398	EIBD398	0000 0001$_H$	32	FFFF BE38$_H$	0	32
INTC2	EI Level Interrupt Bind Register 399	EIBD399	0000 0001$_H$	32	FFFF BE3C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 400	EIBD400	0000 0001$_H$	32	FFFF BE40$_H$	0	32
INTC2	EI Level Interrupt Bind Register 401	EIBD401	0000 0001$_H$	32	FFFF BE44$_H$	0	32
INTC2	EI Level Interrupt Bind Register 402	EIBD402	0000 0001$_H$	32	FFFF BE48$_H$	0	32
INTC2	EI Level Interrupt Bind Register 403	EIBD403	0000 0001$_H$	32	FFFF BE4C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 404	EIBD404	0000 0001$_H$	32	FFFF BE50$_H$	0	32
INTC2	EI Level Interrupt Bind Register 405	EIBD405	0000 0001$_H$	32	FFFF BE54$_H$	0	32
INTC2	EI Level Interrupt Bind Register 406	EIBD406	0000 0001$_H$	32	FFFF BE58$_H$	0	32
INTC2	EI Level Interrupt Bind Register 407	EIBD407	0000 0001$_H$	32	FFFF BE5C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 408	EIBD408	0000 0001$_H$	32	FFFF BE60$_H$	0	32
INTC2	EI Level Interrupt Bind Register 409	EIBD409	0000 0001$_H$	32	FFFF BE64$_H$	0	32
INTC2	EI Level Interrupt Bind Register 410	EIBD410	0000 0001$_H$	32	FFFF BE68$_H$	0	32
INTC2	EI Level Interrupt Bind Register 411	EIBD411	0000 0001$_H$	32	FFFF BE6C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 412	EIBD412	0000 0001$_H$	32	FFFF BE70$_H$	0	32
INTC2	EI Level Interrupt Bind Register 413	EIBD413	0000 0001$_H$	32	FFFF BE74$_H$	0	32
INTC2	EI Level Interrupt Bind Register 414	EIBD414	0000 0001$_H$	32	FFFF BE78$_H$	0	32
INTC2	EI Level Interrupt Bind Register 415	EIBD415	0000 0001$_H$	32	FFFF BE7C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 416	EIBD416	0000 0001$_H$	32	FFFF BE80$_H$	0	32

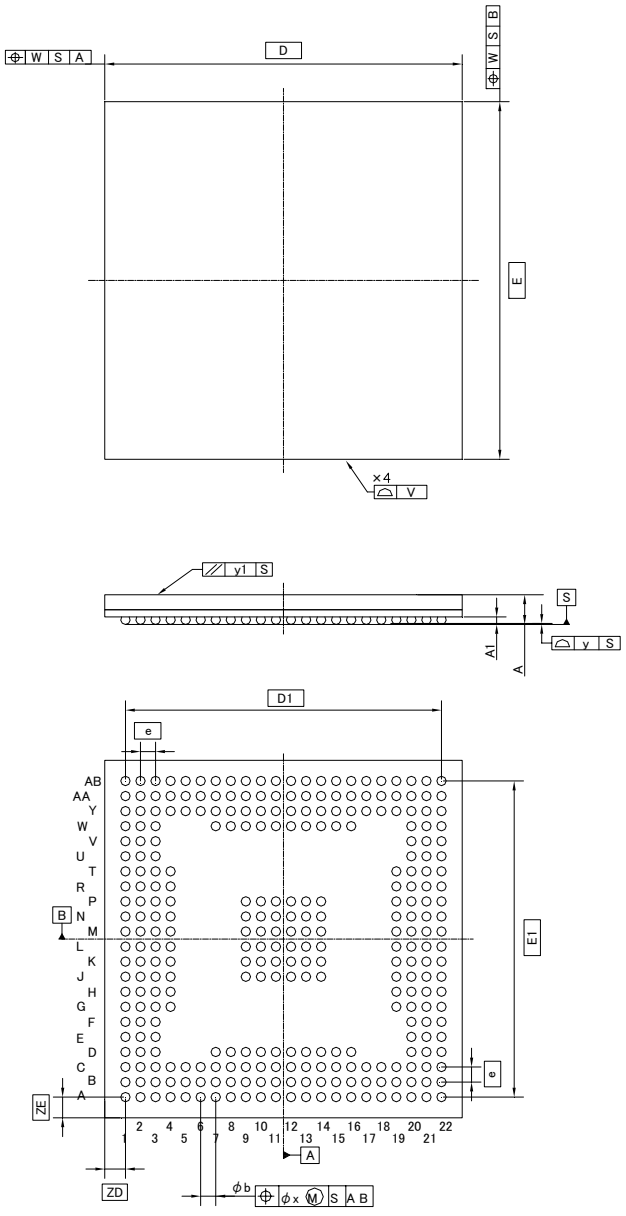
Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
INTC2	EI Level Interrupt Bind Register 417	EIBD417	0000 0001$_H$	32	FFFF BE84$_H$	0	32
INTC2	EI Level Interrupt Bind Register 418	EIBD418	0000 0001$_H$	32	FFFF BE88$_H$	0	32
INTC2	EI Level Interrupt Bind Register 419	EIBD419	0000 0001$_H$	32	FFFF BE8C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 420	EIBD420	0000 0001$_H$	32	FFFF BE90$_H$	0	32
INTC2	EI Level Interrupt Bind Register 421	EIBD421	0000 0001$_H$	32	FFFF BE94$_H$	0	32
INTC2	EI Level Interrupt Bind Register 422	EIBD422	0000 0001$_H$	32	FFFF BE98$_H$	0	32
INTC2	EI Level Interrupt Bind Register 423	EIBD423	0000 0001$_H$	32	FFFF BE9C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 424	EIBD424	0000 0001$_H$	32	FFFF BEA0$_H$	0	32
INTC2	EI Level Interrupt Bind Register 425	EIBD425	0000 0001$_H$	32	FFFF BEA4$_H$	0	32
INTC2	EI Level Interrupt Bind Register 429	EIBD429	0000 0001$_H$	32	FFFF BEB4$_H$	0	32
INTC2	EI Level Interrupt Bind Register 434	EIBD434	0000 0001$_H$	32	FFFF BEC8$_H$	0	32
INTC2	EI Level Interrupt Bind Register 435	EIBD435	0000 0001$_H$	32	FFFF BECC$_H$	0	32
INTC2	EI Level Interrupt Bind Register 436	EIBD436	0000 0001$_H$	32	FFFF BED0$_H$	0	32
INTC2	EI Level Interrupt Bind Register 437	EIBD437	0000 0001$_H$	32	FFFF BED4$_H$	0	32
INTC2	EI Level Interrupt Bind Register 438	EIBD438	0000 0001$_H$	32	FFFF BED8$_H$	0	32
INTC2	EI Level Interrupt Bind Register 439	EIBD439	0000 0001$_H$	32	FFFF BEDC$_H$	0	32
INTC2	EI Level Interrupt Bind Register 440	EIBD440	0000 0001$_H$	32	FFFF BEE0$_H$	0	32
INTC2	EI Level Interrupt Bind Register 441	EIBD441	0000 0001$_H$	32	FFFF BEE4$_H$	0	32
INTC2	EI Level Interrupt Bind Register 442	EIBD442	0000 0001$_H$	32	FFFF BEE8$_H$	0	32
INTC2	EI Level Interrupt Bind Register 443	EIBD443	0000 0001$_H$	32	FFFF BEEC$_H$	0	32
INTC2	EI Level Interrupt Bind Register 444	EIBD444	0000 0001$_H$	32	FFFF BEF0$_H$	0	32
INTC2	EI Level Interrupt Bind Register 445	EIBD445	0000 0001$_H$	32	FFFF BEF4$_H$	0	32
INTC2	EI Level Interrupt Bind Register 446	EIBD446	0000 0001$_H$	32	FFFF BEF8$_H$	0	32
INTC2	EI Level Interrupt Bind Register 447	EIBD447	0000 0001$_H$	32	FFFF BEFC$_H$	0	32
INTC2	EI Level Interrupt Bind Register 448	EIBD448	0000 0001$_H$	32	FFFF BF00$_H$	0	32
INTC2	EI Level Interrupt Bind Register 449	EIBD449	0000 0001$_H$	32	FFFF BF04$_H$	0	32
INTC2	EI Level Interrupt Bind Register 450	EIBD450	0000 0001$_H$	32	FFFF BF08$_H$	0	32
INTC2	EI Level Interrupt Bind Register 451	EIBD451	0000 0001$_H$	32	FFFF BF0C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 452	EIBD452	0000 0001$_H$	32	FFFF BF10$_H$	0	32
INTC2	EI Level Interrupt Bind Register 453	EIBD453	0000 0001$_H$	32	FFFF BF14$_H$	0	32
INTC2	EI Level Interrupt Bind Register 454	EIBD454	0000 0001$_H$	32	FFFF BF18$_H$	0	32
INTC2	EI Level Interrupt Bind Register 455	EIBD455	0000 0001$_H$	32	FFFF BF1C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 456	EIBD456	0000 0001$_H$	32	FFFF BF20$_H$	0	32
INTC2	EI Level Interrupt Bind Register 457	EIBD457	0000 0001$_H$	32	FFFF BF24$_H$	0	32
INTC2	EI Level Interrupt Bind Register 458	EIBD458	0000 0001$_H$	32	FFFF BF28$_H$	0	32
INTC2	EI Level Interrupt Bind Register 459	EIBD459	0000 0001$_H$	32	FFFF BF2C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 460	EIBD460	0000 0001$_H$	32	FFFF BF30$_H$	0	32
INTC2	EI Level Interrupt Bind Register 461	EIBD461	0000 0001$_H$	32	FFFF BF34$_H$	0	32
INTC2	EI Level Interrupt Bind Register 462	EIBD462	0000 0001$_H$	32	FFFF BF38$_H$	0	32
INTC2	EI Level Interrupt Bind Register 463	EIBD463	0000 0001$_H$	32	FFFF BF3C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 464	EIBD464	0000 0001$_H$	32	FFFF BF40$_H$	0	32
INTC2	EI Level Interrupt Bind Register 465	EIBD465	0000 0001$_H$	32	FFFF BF44$_H$	0	32
INTC2	EI Level Interrupt Bind Register 466	EIBD466	0000 0001$_H$	32	FFFF BF48$_H$	0	32
INTC2	EI Level Interrupt Bind Register 467	EIBD467	0000 0001$_H$	32	FFFF BF4C$_H$	0	32
INTC2	EI Level Interrupt Bind Register 468	EIBD468	0000 0001$_H$	32	FFFF BF50$_H$	0	32
INTC2	EI Level Interrupt Bind Register 469	EIBD469	0000 0001$_H$	32	FFFF BF54$_H$	0	32
INTC2	EI Level Interrupt Bind Register 470	EIBD470	0000 0001$_H$	32	FFFF BF58$_H$	0	32

Module Name	Register Name	Register Symbol	Value After Reset	Number of Bits	Address	Peripheral Group	Access Size
INTC2	EI Level Interrupt Bind Register 471	EIBD471	0000 0001_H	32	FFFF $BF5C_H$	0	32
INTC2	EI Level Interrupt Bind Register 472	EIBD472	0000 0001_H	32	FFFF $BF60_H$	0	32
INTC2	EI Level Interrupt Bind Register 473	EIBD473	0000 0001_H	32	FFFF $BF64_H$	0	32
INTC2	EI Level Interrupt Bind Register 474	EIBD474	0000 0001_H	32	FFFF $BF68_H$	0	32
INTC2	EI Level Interrupt Bind Register 475	EIBD475	0000 0001_H	32	FFFF $BF6C_H$	0	32
INTC2	EI Level Interrupt Bind Register 480	EIBD480	0000 0001_H	32	FFFF $BF80_H$	0	32
INTC2	EI Level Interrupt Bind Register 481	EIBD481	0000 0001_H	32	FFFF $BF84_H$	0	32
INTC2	EI Level Interrupt Bind Register 482	EIBD482	0000 0001_H	32	FFFF $BF88_H$	0	32
INTC2	EI Level Interrupt Bind Register 483	EIBD483	0000 0001_H	32	FFFF $BF8C_H$	0	32
INTC2	EI Level Interrupt Bind Register 484	EIBD484	0000 0001_H	32	FFFF $BF90_H$	0	32
INTC2	EI Level Interrupt Bind Register 485	EIBD485	0000 0001_H	32	FFFF $BF94_H$	0	32
INTC2	EI Level Interrupt Bind Register 486	EIBD486	0000 0001_H	32	FFFF $BF98_H$	0	32
INTC2	EI Level Interrupt Bind Register 487	EIBD487	0000 0001_H	32	FFFF $BF9C_H$	0	32
INTC2	EI Level Interrupt Bind Register 489	EIBD489	0000 0001_H	32	FFFF $BFA4_H$	0	32

Appendix Package Dimensions

JEITA Package code	RENESAS code	Previous code	MASS(TYP.)[g]
P-FBGA304-19x19-0.80	PRBG0304GB-A	-	1.1

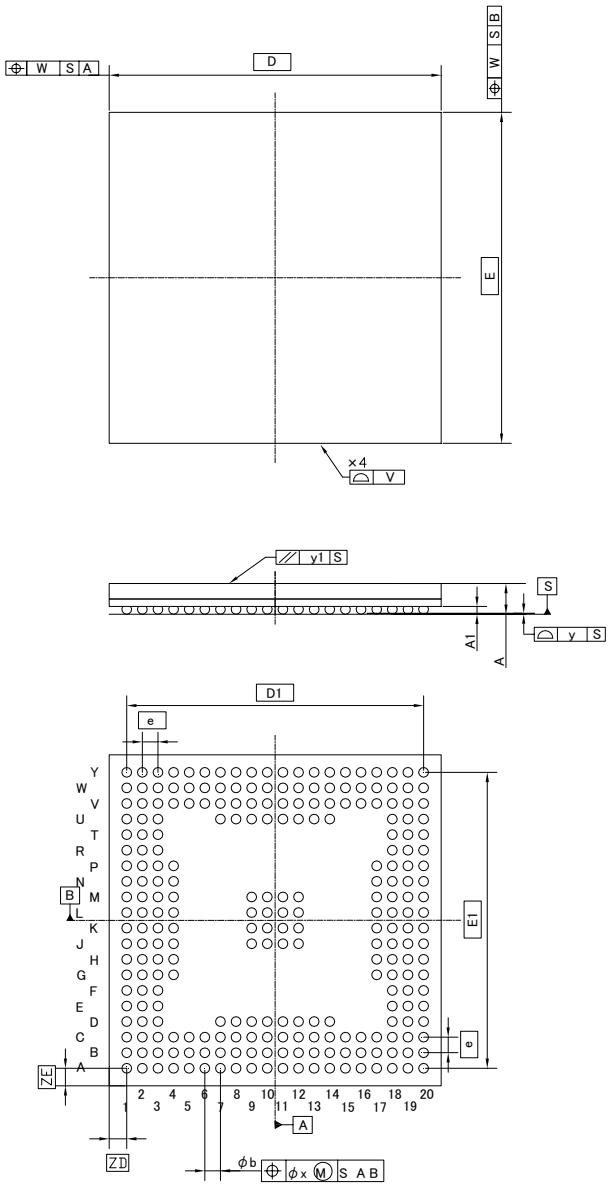
Unit:mm



Referance Symbol	Dimension in Millimeters		
	Min	Mon	Max
D	—	19.00	—
D1	—	16.80	—
E	—	19.00	—
E1	—	16.80	—
v	—	—	0.15
w	—	—	0.20
e	—	0.80	—
A	—	1.58	2.00
A1	0.30	0.35	0.40
b	0.49	0.54	0.59
x	—	—	0.08
y	—	—	0.10
y1	—	—	0.20
ZD	—	1.10	—
ZE	—	1.10	—

JEITA Package code	RENESAS code	Previous code	MASS(TYP.)[g]
P-FBGA252-17x17-0.80	PRBG0252GB-A	-	0.90

Unit:mm



Referance Symbol	Dimension in Millimeters		
	Min	Mon	Max
D	—	17.00	—
D1	—	15.20	—
E	—	17.00	—
E1	—	15.2	—
v	—	—	0.15
w	—	—	0.20
e	—	0.80	—
A	—	1.58	2.00
A1	0.30	0.35	0.40
b	0.49	0.54	0.59
x	—	—	0.08
y	—	—	0.10
y1	—	—	0.20
ZD	—	0.90	—
ZE	—	0.90	—

REVISION HISTORY	RH850/E1M-S User's Manual: Hardware

Rev.	Date	Description	
		Page	Summary
1.00	Apr 21, 2015	—	First Edition issued
1.10	Apr 15, 2016	Section 1 Overview	
		71	Table 1.2 Pin Numbers and Pin Names (E1MS-BGA252) (6/6), "/" for W12 deleted
		72	Figure 1.3 Internal Block Diagram, description added
		Section 2 Pins	
		160	Table 2.74 Pin State (1/2), P7 and P8 pins modified, P9 and P12 deleted
		163	Table 2.75 Examples of the Handling of Unused Pins (2/4), description of "on-chip pull-down/pull-up resister" of "AUD RAM monitor" and "Debug" modified
		Section 3 CPU System	
		166	Figure 3.1 Block Configuration Diagram, modified (an arrow from "PCU (PE3") to "P-Bus" added)
		167	3.1.1 Block Configuration, description of P-Bus and H-Bus modified
		188	Table 3.32 FPIPR Register Contents, description of the FPIPR bit modified
		190	Table 3.34 PMR Register Contents, description of the PM15-0 bits modified
		251	Table 3.109 MCTL Register Contents, a note added to the functional description of the bit, description in Note 1 modified
		253	Table 3.112 PMR Register Contents, description of the PM7-0 bits modified
		267, 268	3.4.3.2 Operation of the LDL.W and STC.W Instructions, description modified, Note added
		269	3.5.1.1 When updated results in the control registers are reflected in the implementation of a subsequent instruction, description modified
		Section 6 Interrupt	
		279	6.2.1 Register Configuration, description modified
		290	6.2.10 PINT0 to PINT7, PINTCLR0 to PINTCLR7 — Peripheral Interrupt Status Registers and Peripheral Interrupt Status Clear Registers, description modified
		298	6.4 Interrupt Exception Handler and Priority Operations, CAUTION added
		322	Table 6.18 Interrupt Response Times, Note, Iϕ = CLK_CPU added, and modified "is" to "I$"
		Section 7 DMA	
		330	Figure 7.4 DTS Channel Arbitration, DMA transfer request cycles for channel 1 modified
		331	7.2.2.3 Interface Arbitration, description added
		341	7.3.1 Suspending and Resuming DMA Transfer by Software Control, description modified
		341	7.3.2 Suspending, Resuming, and Aborting Transfer by a DMAC Channel, description modified
		342, 343	7.3.3 Suspending, Resuming, and Aborting Transfer by the DTS, description modified
		344	Table 7.5 List of Suspend, Resume, and Transfer Abort Functions, description under "How to execute the function" modified
		348	Table 7.6 Master Information That Is Output from DMA, VCID deleted
		349	7.5.4.1 Restriction on the Next Channel in the Chain, description modified (description of VCID deleted)
		365	Table 7.11 DMACTL Register Contents, CAUTION added
		366	Table 7.12 DTSCTL1 Register Contents, CAUTION added
		373	Table 7.19 DM0CMV Register Contents, functional description of the MINF[6:1] bits modified
		374	Table 7.20 DM1CMV Register Contents, functional description of the MINF[6:1] bits modified
		375	Table 7.21 DTSCMV Register Contents, functional description of the MINF[6:1] bits modified
		385	Table 7.35 DTRTRDAT Register Contents, functional description of the TRDAT[6:0] bits modified
		387	7.9.2.20 DTSnnnCM — DTS Channel Master Setting Register (nnn = 000 to 127), the recommended procedure for setting the DTS channel master setting register, functional description of the bits modified
		393	Table 7.42 DTCTn Register Contents (1/3), functional description of bit 27 modified (DTSTn → DCSTn)
		Section 11 Clock Controller	
		453	Table 11.1 List of Clocks, ASIC clock added
		454	Figure 11.1 Block Diagram of Clock Controller, ASIC clock added

Rev.	Date	Description	
		Page	Summary
1.10	Apr 15, 2016	458, 459	Table 11.6 PLL0CLKC1 Register Contents, functional description of bits 7 to 3 and 2 to 0 modified
		466	11.4.10 PROT1PHCMD — Protect 1 Command Register, Note 1 modified
		475	Table 11.17 ACK0CKC Register Contents, Note 1 modified
		Section 12 Standby Controller	
		491	12.3.3 PROT0PHCMD — Protection Command Register, Note 1 modified
		Section 13 Clocked Serial Interface H (CSIH)	
		517	13.5.1 (2) Slave mode, NOTE modified
		556	13.6.3 CSIHnCTL2 — CSIH Control Register 2, description modified
		564	Table 13.23 CSIHnCFGx Register Contents (3/4), table in the functional description of bits 14 to 12 modified
		Section 14 Serial Communication Interface 3 (SCI3)	
		588	Table 14.11 Relationship between Setting N in SCI3nBRR and Bit Rate B, Note modified
		589	14.3.10 SCI3nBRR — Bit Rate Register, description modified
		591	Table 14.17 Relationship between SCI3nMDDR Setting and Bit Rate B when Bit Rate Modulation Function Is Used, Note modified
		595	14.4.3 Clock, description modified
		595	14.4.4 Double-Speed Operation, description modified
		596	Figure 14.5 Sample Flowchart for SCI3 Initialization, description in [5] modified
		599	Figure 14.8 Example Flowchart for Stopping the SCI3 after Serial Transmission, [6] and Note 1 modified
		602	Figure 14.10 Example of Serial Receive Flowchart (1), modified
		607	Figure 14.15 Example of Multi-Processor Serial Receive Flowchart (1), [1] modified
		608	Figure 14.16 Example of Multi-Processor Serial Receive Flowchart (2), Note 1 modified
		614	Figure 14.22 Example Flowchart for Stopping the SCI3 after Serial Transmission, Note 1 modified
		616	Figure 14.24 Example of Serial Receive Flowchart, description modified
		618	Figure 14.25 Example of Simultaneous Serial Transmission and Receive Flowchart, modified
		622	14.9.6 External Clock Input in Clock Synchronous Mode, description modified
		Section 16 CAN Interface (RS-CAN)	
		678	Table 16.8 External I/O Signals, title modified
		678	Table 16.8 External I/O Signals, description of Txm (m = 0 to 3) modified
		680	Table 16.9 RS-CAN Module Specifications (2/2), description of "interrupt source" modified (transmit mode → receive mode)
		862	16.6 Notes on the RS-CAN Module, modified
		Section 18 Renesas High-Speed Bus (RHSB)	
		1171	18.2.5.1 RHSBjUCR — Upstream Configuration Register, R/W of bits 12 to 8 modified to "R"
		Section 19 Window Watchdog Timer (WDTA)	
		1254	19.5.1.3 WDTA settings after reset release, setting of 75% interrupt mode in the table modified
		Section 20 OS Timer (OSTM)	
		1261	20.1 Functional Overview, description modified
		1272	Figure 20.4 Flow of Starting the OS Timer, description in Note 1 modified
		Section 21 Advanced Timer Unit IV (ATU-IV)	
		All	In the ATU-IV section, pin and signal names have been modified as follows: TOCxy → TIOCxy, TIA0 to 6 → TIA00 to 06
		1328	21.1.4 Clock Supply, added
		1347	Table 21.10 TCR2A Register Contents, entries modified (TIAx → TIA00 to 06)
		1358, 1359	Table 21.18 TIOR2A Register Contents, entries modified (TIAx → TIA00 to 06)
		1372	Table 21.26 NCRAx Register Contents, entries modified (TIAx → TIA0x)
		1377	Figure 21.11 Example of Free-Running Counter A (TCNTA) Operation: Overflow Timing, values of TCNTA modified
		1437	Figure 21.21 Compare-Match Operation of TCNTB6 and Output of CMFB6 Interrupt (IREGB6 = 00, 01), description modified

Rev.	Date	Description	
		Page	Summary
1.10	Apr 15, 2016	1437	Figure 21.22 CMFB6 Interrupt Output when IREGB6 = 10, Note modified
		1488	Figure 21.39 One-Shot Pulse Operation (Output Pin Initialization Timing Chart), pin names modified (toc, tocx[0] → TIOCxy)
		1490	Figure 21.42 One-Shot Pulse Output Cancellation Timing Chart, pin name modified (at_tioc00 → TIOC00)
		1495	21.7.1 Operation Overview, description modified (TOD0xA, TOD0xB → TOD0yA, TOD0yB)
		1502, 1503	21.7.2.3 TIOR1Dx — Timer I/O Control Registers 1Dx, description modified
		1504	21.7.2.4 TIOR2Dx — Timer I/O Control Register 2Dx, description modified (TODxA → TODxyA)
		1504	21.7.2.4 (1) IOBDxy[2:0] — I/O Control B, description modified (TODxA → TODxyA)
		1574, 1575	21.9.1 Operation Overview, description modified
		1575	Figure 21.59 Block Diagram of Subblocks of Timer F, signal name modified, Note 1 added
		1580	Table 21.112 NCMCR2F Register Contents, description modified
		1582	Table 21.114 NCCRF Register Contents, description modified
		1583	21.9.2.4 (1) NCEFx — Noise Canceller Enable Fx, description modified
		1585	Table 21.116 TCR1Fx Register Contents, functional description of bits 1 and 0 modified
		1586	21.9.2.6 (3) EGSELFx — Edge Select Fx, description modified
		1587	Table 21.117 TCR2Fx Register Contents, functional description of bits 7 to 5 modified
		1587	21.9.2.7 TCR2Fx — Timer Control Registers 2Fx, description modified
		1597	21.9.2.13 ECNTBFx — Event Counters Fx, description modified
		1597	Table 21.124 Event Counter Input Clocks and Count Edges for Each Operation Mode of Timer, input clock modified
		1602	21.9.2.18 GRCFx — General Registers CFx, description modified (TIFnA → TIFx)
		1608, 1609	21.9.2.23 NCNTFAx — Noise Canceler Counters FAx, description modified
		1612	Table 21.136 NCRFAx Register Contents, functional description modified
		1614	21.9.3.1 Edge Counting in a Given Time, description of ECNTBFx modified
		1616	21.9.3.2 Effective Edge Interval Counting, description modified
		1618	21.9.3.3 Measurement of Time during High/Low Input Levels, description modified
		1620	21.9.3.4 Measurement of PWM Input Waveform Timing, description modified
		1623	21.9.3.5 Rotation Speed/Pulse Measurement, description modified
		1624	Figure 21.67 Operation Example of Rotation Speed/Pulse Measurement, signal name modified
		1646	21.11.2.6 TCNT2H — Timer Counter 2H, description modified
		Section 22 Autonomous Pulse Adapter (APA)	
		1821	Table 22.65 Standard Setting and Operation Procedure, description of step 1 modified
		1822	Table 22.66 Setting Example for Avoiding Unintended Exception Processing, entries in the table and Note modified
		Section 23 Motor Control Timer (TSG2)	
		1839	Figure 23.1 Block Diagram of TSG2n, modified
		1840, 1841	Table 23.5 TSG2n (n = 0, 1) Registers, modified
		1848	Table 23.11 TSG2nCTL6 Register Contents (1/2), description of bits 7 and 4 modified (STADT0/1 → TSTADT0/1)
		1848	Table 23.11 TSG2nCTL6 Register Contents (1/2), description of bits 9 to 6 modified (TSTADT0 → TSTADT1)
		1893	Figure 23.8 Anytime Rewriting Timing (Example in PWM Mode), interrupt names modified
		1934	Figure 23.43 Example of Error Interrupt (INTTSG2nIER) Generation (PWM Mode), pin names modified (TSG2n01 and TSG2n02 pins → TSOn1 and TSOn2 pins)
		1934	Figure 23.44 Example of Error Interrupt (INTTSG2nIER) Generation for Each Active Level, pin names modified (TSG2n01 and TSG2n02 pins → TSOn1 and TSOn2 pins)
		1935	Figure 23.45 Example of Error Interrupt Operation, pin names modified
		1939	Table 23.47 Timer Output in PWM Mode, entry under "Pin" modified (TSG2nOm → TSOnm)
		1940	Table 23.50 Example of Setting Each Timer Output Condition in PWM Mode, entry under "Pin" modified (TSG2nOm → TSOnm)
		1944	23.11.1.3 Controlling Dead Time in PWM Mode, description modified (TSG2n02 → TSOn2)

Rev.	Date	Description	
		Page	Summary
1.10	Apr 15, 2016	1980	23.11.4 120-DC Mode, Overview, modified (two types → three types)
		1980	Figure 23.70 Block Diagram in 120-DC Mode, modified
		1982	Table 23.66 Timer Input Function in 120-DC Mode, modified
		1983	Table 23.70 Example of Setting Each Timer Output Condition in 120-DC Mode, entry under "Pin" modified
		1987	23.11.4.3 Control Methods in 120-DC Mode, entries in the table modified
		1988	Operation of software output control method, description modified
		1992	Figure 23.75 Example of Operation in 120-DC Mode (Normal Rotation: TSG2nSTR1.TSG2nTSF = 0 and TSG2nOPT0.TSG2nIDC = 1), modified
		1993	Figure 23.76 Example of Operation in 120-DC Mode (Reverse Rotation: TSG2nSTR1.TSG2nTSF = 1 and TSG2nOPT0.TSG2nIDC = 0), modified
		Section 25 Peripheral Interconnection (PIC)	
		2026	25.1.2.3 PIC1SSER2 — Simultaneous Start Control Register 2, description modified
		2027	25.1.2.4 PIC1SSER3 — Simultaneous Start Control Register 3, description modified
		2028	25.1.2.5 PIC1HIZCEN2 — Hi-Z Output Control Register 2, description modified
		2028	Table 25.7 PIC1HIZCEN2 Register Contents, functional description of bit 0 modified
		2029	25.1.2.6 PIC1HIZCEN3 — Hi-Z Output Control Register 3, description modified
		2029	Table 25.8 PIC1HIZCEN3 Register Contents, functional description of bit 0 modified
		2030	25.1.3.1 Timer Synchronization, description in (1) modified
		2037	25.2.3.2 PIC2ADCBnEDGSEL — AD Converter n Trigger Edge Selection Control Register, description modified
		2056, 2057	Table 25.22 PIC2DSADCATSEL0 Register Contents, functional description of bits 15, 13, 11, 9, 7, 5, 3, and 1 modified (ADTRG_DS7 to 0 → DSADTRG7 to 0)
		2058	Table 25.23 PIC2DSADCATSEL1 Register Contents (1/2), description of bit 15 modified (DSADC6 → DSADC7)
		2058, 2059	Table 25.23 PIC2DSADCATSEL1 Register Contents, description of bits 15, 13, 11, 9, 7, 5, 3, and 1 modified (ADTRG_DS7 to 0 → DSADTRG7 to 0)
		2060	25.2.4.1 ADC Trigger Selection Function, description modified
		Section 26 A/D Converter (ADCB)	
		2069	Table 26.4 Register Address List (1/2), access size of data register n modified
		2074	Table 26.7 ADCBmVCRn Register Contents (2/2), functional description of bits 5 to 0 modified
		2076	26.5.2 ADCBmDRn — Data Register n, description modified
		2089	Table 26.20 ADCBmODCR Register Contents, description in Note modified
		2098	Table 26.30 ADCBmSGCRx Register Contents (x = 0 to 2), functional description of bit 0 modified
		2099	Table 26.31 ADCBmSGCRx Register Contents (x = 3 or 4), functional description of bits 1 and 0 modified
		2107	Figure 26.3 Initial Settings, modified (ADCBmADTIRR3-4 → ADCBmADTIPR3-4)
		2121, 2122	26.7.9.2 A/D Converter Self-Diagnosis, description and figure added
		2123	26.7.10 Sampling of an Analog Input and Processing Time for a Scan Group, the calculation formula of the processing time modified
		2123	Table 26.40 Processing Time for a Scan Group, description added
		2135	26.9.3 Notes when Current is Being Injected, added
		Section 28 Digital Filter (DFE)	
		2189	Table 28.2 List of Terms (2/2), description of timer trigger modified
		2193	Table 28.3 List of Control Registers (4/4), entries under "Function" modified
		2231	28.4.7.2 Trigger Flag Functions, modified (CTRLACHn.EN → CTLACHn.EN)
		2259	Table 28.32 Processing Time of FIR and IIR (Normal Cases), time modified
		2259	Table 28.33 Processing Time of FIR and IIR (with Initialization), time modified
		2259	28.5.3 DFE Processing Time, description modified
		2262	28.5.8 Operation when a Channel is Disabled, modified (CTRLACHn.EN → CTLACHn.EN)

Rev.	Date	Description	
		Page	Summary
1.10	Apr 15, 2016	Section 29 Safety	
		2267	Table 29.3 Overview of Code Flash ECC, description of self-diagnosis modified
		2275	29.2.2.3 (6) CF1STERSTR_VCI/_PE1/_PCU — Code Flash 1st Error Status Register, description modified
		2275	Table 29.10 CF1STERSTR Register Contents, description of bits 2 to 0 modified
		2282	Table 29.18 DFERSTR Register Contents, description of bits 1 and 0 modified
		2288	Table 29.25 Overview of Local RAM ECC of CPU1, "Address capture" deleted
		2290	Table 29.28 List of Registers, modified (Local RAM 1st error address register n deleted
		2292	Table 29.30 LRTSTCTL_PE1 Register Contents, description of bits 5 to 2 modified
		2293	29.2.4.3 (3) LRTDATBFn_PE1 — Local RAM Test Data Read Buffer n (n = 0, 1), bit chart modified
		2293	Table 29.31 LRTDATBFn_PE1 Register Contents, table title and bit name modified
		2296	29.2.4.3 (6) LRSTCLR_PE1 — Local RAM Status Clear Register, description modified
		2296	Table 29.34 LRSTCLR_PE1 Register Contents, functional description of bits 3 to 0 modified
		2298	29.2.4.3 (8) LR1STERSTR_PE1 — Local RAM 1st Error Status Register, description modified
		2298	Table 29.36 LR1STERSTR_PE1 Register Contents, functional description modified
		—	29.2.4.3 (9) LR1STEADRn_PE1 — Local RAM 1st Error Address Register n (n = 0 to 3), deleted
		2304	29.2.5.3 (2) LRFSTERSTR_PCU — Local RAM 1st Error Status Register, description modified
		2304	Table 29.40 LRFSTERSTR_PCU Register Contents, description of bits 1 and 0 modified
		2306	Table 29.42 LROVFSTR_PCU Register Contents, description of bit 0 modified
		2310	Table 29.46 LRTSTCTRL_PCU Register Contents, description of bits 1 and 0 modified
		2311	Table 29.47 LRTDATBF0_PCU Register Contents, bit names and description of bits 6 to 0 modified
		2316, 2317	Table 29.52 GRTSTCTL Register Contents, description of bits 7 to 4 and 2 to 0 modified
		2318	Table 29.53 GRTDATBFn Register Contents, Note added
		2325	29.2.6.3 (10) GR1STERSTR_VCI/PE1 — Global RAM 1st Error Status Register, description modified
		2325, 2326	Table 29.60 GR1STERSTR_VCI/PE1 Register Contents, entries under bit position and functional description modified
		2331	Table 29.62 Overview of Instruction Cache ECC, description of ECC error detection and correction modified
		2337	29.2.7.3 (5) ID1STERSTR_PE1 — Instruction Cache Data RAM 1st Error Status Register, description modified
		2337	Table 29.68 ID1STERSTR_PE1 Register Contents, description of bits 9, 8, 1, and 0 modified
		2338	29.2.7.3 (6) ID1STEADRn_PE1 — Instruction Cache Data RAM (Bank n) 1st Error Address Register (n = 0, 1), description modified
		2342	Table 29.73 ITOVFSTR_PE1 Register Contents, entries under bit position modified
		2343	29.2.7.3 (11) IT1STERSTR_PE1 — Instruction Cache Tag RAM 1st Error Status Register, description modified
		2343	Table 29.74 IT1STERSTR_PE1 Register Contents, description of bits 1 and 0 modified
		2344	29.2.7.3 (12) IT1STEADR0_PE1 — Instruction Cache Tag RAM 1st Error Address Register, description modified
		2379	29.3 Lockstep, description modified
		2389	Table 29.103 MGDGRSSTAT_VCI/PE1 Register Contents, description of bit 1 modified
		2394	Table 29.110 FSGDxxDPROTn Register Contents (1/2), description of bit 4 modified
		2396	Table 29.113 ERRSLVxxADDR Register Contents, functional description modified
		2397	Table 29.114 ERRSLVxxTYPE Register Contents, functional description modified
		2398	Table 29.115 Specification Overview, description of signature generation modified
		2404	29.5.3.3 Data Counter, CAUTIONS, description modified
		2405	Table 29.117 List of Registers of the Signature Generation Units, RFU, value after reset modified

REVISION HISTORY	RH850/E1M-S User's Manual: Hardware

Rev.	Date	Description	
		Page	Summary
1.10	Apr 15, 2016	2423	Table 29.133 List of Clocks Monitored by Each Clock Monitor and Sampling Clocks Used, description modified
		Section 30 Error Control Module (ECM)	
		2440	30.1.3 Operations for ERROROUT Output, description modified (after release from the reset state → in the reset state or after release from the reset state)
		Section 33 Flash Memory	
		2522	33.11 Usage Notes, (4) and (5) modified
		Section 35 RAM	
		2531	Figure 35.1 Circuit Configuration around the Emulation RAM, modified
		Section 36 Boundary Scan	
		2543	36.3 Register Descriptions, description modified
		2547	Table 36.5 Pins not Subjected to Boundary Scan, description of analog input modified
		2548	36.5 Usage Notes, 6., description modified
		Section 37 Electrical Characteristics	
		2549	Table 37.1 Absolute Maximum Ratings, Rated value of EVCC power pins modified
		2559	Table 37.4 DC Characteristics (Input Voltage), max. value of VT+ modified
		2559	Table 37.4 DC Characteristics (Input Voltage), notes added
		2562	Table 37.7 DC Characteristics (Output Voltage), min. value of EVCC power pin modified
		2567	Table 37.13 DC Characteristics (Standby), conditions modified
		2567	Table 37.14 DC Characteristics (Power Voltage Monitoring), conditions modified
		2569	Table 37.15 Power On/Off Timings, specification of PLL1 lock-in time modified
		2570	Table 37.16 Power On/Off Timings, specification of PLL1 lock-in time modified
		2577	Table 37.26 CSIH Timing in Master Mode, item added, entries modified
		2578	Figure 37.12 CSIH Timing (Master Mode), condition added
		2581	Table 37.27 CSIH Timing in Slave Mode, items added
		2582	Figure 37.16 CSIH Timing (Slave Mode), condition added
		2583	Figure 37.17 CSIH Timing (Slave Mode), item added
		2584	Figure 37.18 CSIH Timing (Slave Mode), pin names modified
		2585	Term unified (SCI → SCI3)
		2585	Table 37.28 SCI3 Timing (Master Mode), specification of transmit data delay time modified
		2585	Figure 37.20 SCI3 Input/Output Timing / Clock Synchronous Mode, modified
		2586	Table 37.29 SCI3 Timing (Slave Mode), specification of transmit data delay time and notes modified
		2591	Table 37.34 JTAG/NEXUS Timing, Max. value of tTDOD and tRDYD modified, name of the DRDY pin modified (RDY → DRDY)
		2591	Figure 37.25 JTAG/NEXUS Timing, name of the DRDY pin modified (RDY → DRDY)
		2594	Table 37.37 ADC Converter Characteristics, Self-diagnosis absolute error modified, CAUTION added
		2595	37.4 A/D Converter Characteristics, • Errors in the External Circuit of the A/D Convertor, items in the table modified
		2596	Table 37.38 ΔΣADC Conversion Characteristics (1/2), conditions in examples 1 to 4 modified
		2598	Table 37.40 Code Flash Basic Characteristics, Note 2, description modified
		2599	Table 37.43 Data Flash Programming Characteristics, unit of block size modified (4 B → 4 KB)
		Appendix List of Registers	
		2601 to 2801	Entries in the register table modified, "Peripheral Group" column added to the table
1.20	May 31, 2017	How to Use This Manual	
		6	3. Register Notation, figure modified and description added to "R:" in (4).
		Table of Contents	
		7	Table of Contents, spelling error corrected (Table of Contens → Table of Contents)
		Section 1 Overview	
		52	1.2 Features, description of interrupts/exception modified ((ECM) added to 1 FE level interrupt)

Rev.	Date	Description	
		Page	Summary
1.20	May 31, 2017	53	1.2 Features, description of advanced timer unit IV (ATU-IV) modified (Timer H: 32-bit compare match timer → Timer H: 32-bit interval timer)
		54	1.2 Features, description of FlexRay modified
		55	1.2 Features, description of boundary scan added
		72	Figure 1.3 Internal Block Diagram, modified
		Section 2 Pins	
		99	2.1.4.6 Pin-Unit Register, (1) PCRn_m — Port Control Register, CAUTIONS added
		Section 3 CPU System	
		169	Table 3.1 Features of the RH850G3M Core, description modified (Thirty-two 32-bit general registers→Thirty-two 32-bit general-purpose registers)
		199	(8) MCR — Memory protection setting check result register, description modified (Be sure to clear bit 31 to 9, 7, and 6 to 0.→Be sure to clear bit 31 to 9, 7, and 6.)
		222	3.2.4.2, (4) Register Set, (d) IPGPMTUM0 — Peripheral Device Protection Setting Register 0, description in a bit chart modified (Bit → R/W)
		225	Table 3.74 Register Contents of IPGPMTUM4, functional description of W0 and R0 bits modified
		227	Table 3.76 SEGCONT Register Contents (1/2), description of bit 15, 9, 6, and 5 modified
		227	Table 3.76 SEGCONT Register Contents (1/2), functional description of VCRE bit modified (note 2 added)
		228	Table 3.76 SEGCONT Register Contents (2/2) description of bit 4 modified
		228	Table 3.76 SEGCONT Register Contents (2/2), functional description of VCIE bit modified and Note 2 added
		231	3.2.4.3 System Error Notification Control Function (SEG), (c) SEGADDR — Error Factor Retention Register (Address), description added
		234	Table 3.80 Features of the RH850G3K Core, description modified (Thirty-two 32-bit general registers→Thirty-two 32-bit general-purpose registers)
		271	3.5.1 Synchronization of Store Instruction Completion and Subsequent Instruction Generation, body modified
		272	3.5.1.2 When the updated results in the control registers and memories are reflected in the instruction fetch of a subsequent instruction:, body modified
		272	3.5.1.3 When switching the code flash area, added
		272	3.5.1.4 When executing the SYNCM instruction to wait for the completion of update by the store instruction, added
		Section 6 Interrupt	
		285	Table 6.5 EIC Register Contents (2/2), functional description of EIMKn bit modified
		305	Table 6.14 Interrupt Exception Handler and Priority (4/13), functional module of TCNTC7 overflow interrupt (113) modified (ATU-IV timer D → ATU-IV timer C)
		325	Table 6.18 Interrupt Response Times (min.), title of the table modified, entries under "Operating Clock", "Synchronization", and "Total" modified, and Note 1 added
		325	6.6 Interrupt Response Times, table below table 6.18, entries under "In CPU1 (240MHz)" modified, header of the table modified (In PCU (320 MHz)), "In PCU (240 MHz)" column added
		Section 7 DMA	
		329	7.2.1.2 Executing a DMA Cycle, description added
		345	7.3.2 Suspending, Resuming, and Aborting Transfer by a DMAC Channel, description added
		349	Table 7.5 List of Functions for Suspending, Resuming, and Aborting Transfer, Note 3 added
		350	7.4.2 DMA Transfer Error, description added
		353	7.5.2.4 Illegal Access, description added
		393	7.9.2.20 DTSnnnCM — DTS Channel Master Setting Register (nnn = 000 to 127), description modified (Bits 31 to 26 of this register are reserved, →Bits 31 to 23 of this register are reserved,)
		399	Table 7.42 DTCTn Register Contents (2/3), functional description of SACM[1:0] bits modified (setting of SACM1 bit for "Setting prohibited" modified to 1)
		400	Table 7.42 DTCTn Register Contents (3/3), CAUTION 1 modified
		413	Figure 7.12 Structure of the TI, the position where the fields are separated in TI-H modified

Rev.	Date	Description	
		Page	Summary
1.20	May 31, 2017	415	7.11.1.4 Caution about accessing the TI, description modified (Bits 31 to 26 of the TI-H → Bits 31 to 23 of the TI-H)
		Section 8 Resets	
		432	8.3 Reset Sources, entries in the table modified
		437	8.4.6 SWRESA — Software Reset Request Register, Note 1 modified
		Section 9 Power Supply Circuit	
		439	9.1 Features, entry under "Uses of Power Supply" of TTLVCC in the table modified, and the reference added
		Section 10 Power Supply Voltage Monitor	
		455	10.5 Notes on Usage, body modified (DCU deleted)
		Section 11 Clock Controller	
		471	11.4.10 PROT1PHCMD — Protect 1 Command Register, Note 1 modified
		473	11.5.1 Operation When the Divide Function Is Used, 4., description modified (Write 001_B to the CLKD0DIV[2:0] bits→Write 010_B to the CLKD0DIV[2:0] bits)
		474	Figure 11.3 Example of Sequence for Shifting the Clock Gear Up, description of processing 6 modified (100010_B in CKSC1CTL.CKSC[5:0]→100100_B in CKSC1CTL.CKSC[5:0])
		Section 12 Standby Controller	
		491	12.2.1 Power Off Standby Mode, body modified
		495	12.3.3 PROT0PHCMD — Protection Command Register, Note 1 modified
		Section 13 Clocked Serial Interface H (CSIH)	
		526	Figure 13.7 Chip Select and RCB Example, description in the figure modified
		532	Table 13.7 Start of Data Transfer, description of Receive-only modified (Writing to the CSIHnTX0W register on→Writing to the CSIHnTX0W register or)
		534	13.5.7, (2), Example, description modified: 2 to 16 bits plus 8 bits → two 16 bits and 8 bits
		Section 14 Serial Communication Interface 3 (SCI3)	
		590	14.3.8 SCI3nSCMR — Serial Transfer Format Register, R/W of bits 7 to 4, 1, and 0 in a bit chart modified
		591	14.3.9 SCI3nSEMR — Serial Extended Mode Register, R/W of bits 5, 4, 1, and 0 in a bit chart modified
		598	14.4.2 Receive Data Sampling Timing and Receive Margin, description of "N" modified: SABCS → ABCS
		601	14.4.6 Serial Data Transmission (Asynchronous Mode), description added (Supplementary note on operation when data transmission in asynchronous mode is enabled)
		602	Figure 14.6 Example of Operation for Transmission in Asynchronous Mode (Example with 8-Bit Data, Parity, One Stop Bit), description in the figure modified (SCInTDR → SCI3nTDR)
		602	Figure 14.7 Example of Serial Transmission Flowchart, [1] SCI3 initialization, description added
		603	Figure 14.8 Example Flowchart for Stopping the SCI3 after Serial Transmission, description in the figure modified
		604	14.4.7 Serial Data Receive (Asynchronous Mode), description in 5 modified (RDRF flag → SCI3nRDR)
		611	Figure 14.15 Example of Multi-Processor Serial Receive Flowchart (1), description in the figure modified (ORER or RER → Is ORER or FER)
		624	14.8 Interrupt Sources, CAUTION modified
		Section 15 LIN Master Interface (RLIN2)	
		628	Table 15.5 Clock Supply, entries in the table modified
		630	Figure 15.1 LIN Master Interface Block Diagram, pin names modified
		645	Table 15.23 RLN21nmLiMST Register Contents, functional description of the reserved bits modified
		647	15.3.3.10 RLN21nmLiST — LIN Status Register, description of ERR flag (error detection flag) modified (TER → FTER)
		648	15.3.3.11 RLN21nmLiEST — LIN Error Status Register, description of "Access" modified (read → read/written)

Rev.	Date	Description	
		Page	Summary
1.20	May 31, 2017	649	15.3.3.11 RLN21nmLiEST — LIN Error Status Register, description of PBER flag (physical bus error flag) modified (physical error → physical bus error)
		653	15.3.3.14 RLN21nmLiCBR — LIN Checksum Buffer Register, body modified (frame transmission completion→ frame transmission/reception completion)
		669	Table 15.35 Types of Statuses, entry under "Status Set Condition" of "Header reception end" modified
		672	Figure 15.14 Connection in LIN Reset Mode, LIN Wake-Up Mode, and LIN Operation Mode, pin names modified
		673	Figure 15.15 Connection in LIN Self-Test Mode, pin names modified
		678	Figure 15.16 Block Diagram of Baud Rate Generation in LIN Master Mode, Note 3 modified (N → M)
		Section 16 CAN Interface (RS-CAN)	
		684	Table 16.9 RS-CAN Module Specifications (2/2), specification of "Interrupt source" modified (CANm transmit queue interrupt added)
		685	Figure 16.1 RS-CAN Module Block Diagram, incorrect names of interrupts modified
		719	16.3.2.4 RSCAN0CmERFL — Channel Error Flag Register (m = 0 to 3), description of BLE flag modified
		742	16.3.2.18 RSCAN0RMNDy — Receive Buffer New Data Register y (y = 0, 1), bit chart modified
		758	16.3.2.30 RSCAN0CFCCk — Transmit/Receive FIFO Buffer Configuration/Control Register (k = 0 to 11), description of CFITR and CFITSS bits added
		784	16.3.2.45 RSCAN0TMTRSTSy — Transmit Buffer Transmit Request Status Register y (y = 0, 1), bit chart modified
		786	16.3.2.46 RSCAN0TMTARSTSy — Transmit Buffer Transmit Abort Request Status Register (y = 0, 1), bit chart modified
		788	16.3.2.47 RSCAN0TMTCSTSy — Transmit Buffer Transmit Complete Status Register (y = 0, 1), bit chart modified
		790	16.3.2.48 RSCAN0TMTASTSy — Transmit Buffer Transmit Abort Status Register (y = 0, 1), bit chart modified
		792	16.3.2.49 RSCAN0TMIECy — Transmit Buffer Interrupt Enable Configuration Register (y = 0, 1), bit chart modified
		827	Figure 16.6 Entry of Receive Rules (in the case of setting channel 0,1) modified (Page11→Page15)
		843	16.5.1.2 Bit Timing Setting, description added
		845	Figure 16.19 Receive Rule Setting Procedure, description in the figure modified (RSCAN0GAFLCFGy → RSCAN0GAFLCFG0, "y = 0" deleted)
		Section 17 FlexRay	
		867	17.1 FlexRay Module, body modified
		868	17.1.1 Overview, specification of "Communication" modified
		869	17.1.2 Terms and Abbreviations, added
		872	17.1.3 Block Diagram, Timer (TIM), description modified (One absolute timer →Two absolute timers)
		913	Table 17.17 FLXA0FRT2C Register Contents, functional description of T2RC bit modified
		919	Table 17.20 FLXA0FRSUCC1 Register Contents (2/2), functional description of CMD[3:0] bits modified
		933	17.2.6.6 FLXA0FRPRTC2 — FlexRay PRT Configuration Register 2, value after reset of bit 8 modified
		985	17.2.8.4 FLXA0FRFCL — FlexRay FIFO Critical Level Register, value after reset of bit 7 modified
		1033	17.2.12.2 FLXA0FROTC — FlexRay Output Transfer Configuration Register, (6) FLXA0FROTC.OTCS, description modified (NDAT → ND, MBSC bits in the FLXA0FRMBSC register → MBC bits in the FLXA0FRMBSCi register)
		1061	Figure 17.6 Configuration of NIT Start and Offset Correction Start, register name modified
		1086	17.3.9.4 Frame Transmission, register name modified (FLXA0FRTRXQ1 → FLXA0FRTXRQ1)
		1119	17.3.16.1 Input Data Transfer, (3) Input pointer table, Equation 1 modified

Rev.	Date	Description	
		Page	Summary
1.20	May 31, 2017	1129	17.3.16.2 Output Data Transfer, (3) Output pointer table, body modified (NDAT bit → ND bit, MBSC bit → MBC bit)
		1130	17.3.16.2 Output Data Transfer, (5) Transfer function of dedicated message buffers, body modified (FLXA0FRNDAT register → FLXA0FRNDATi register, NDAT flag → ND flag, FLXA0FRMBSC register → FLXA0FRMBSCi register, MBSC flag → MBC flag)
		1140	17.3.16.5 Behaviors in Case of RAM Read Errors, (1) Read error during transfer from TBF to MBF, body modified (FLXA0FRNDAT flag →ND flag in the FLXA0FRNDATi register)
		1143	17.3.16.6 Data Transfer Timings, (2) System bus transfer time, formula of "Equation 8 System bus transfer time" modified
		1147	17.3.17 Byte Alignment, body modified
		Section 18 Renesas High-Speed Bus (RHSB)	
		1162	18.2.4.1 RHSBjDCR — Downstream Configuration Register, (7) Clock Line Phase (RHSBjDCR.RHSBjCLP), description modified (RHSBnFCLN (j = 0, 1)→RHSBjFCLN (j = 0, 1))
		1162	18.2.4.1 RHSBjDCR — Downstream Configuration Register, (8) Clock Active Control (RHSBjDCR.RHSBjCAC), body modified (RHSBjSOP → RHSBjFCLP, RHSBjSON → RHSBjFCLN)
		1205	Figure 18.9 Internal Transmission Processing of Command and Data Frames, description of processing D modified (Assemble command frame →Assemble data frame)
		Section 20 OS Timer (OSTM)	
		1269	20.2.2.2 OSTMnCNT — OSTM Counter Register, Value after reset, description added
		1272	Table 20.10 OSTMnTS Register Contents, functional description of OSTMnTS bit modified (This setting disables the counter.→This setting is invalid.)
		1272	Table 20.11 OSTMnTT Register Contents, functional description of OSTMnTT bit modified (This setting disables the counter.→This setting is invalid.)
		1277	Figure 20.4 Flow of Starting the OS Timer, description in the figure and Note 1 modified (OSTM0TOE → OSTMnTOE, OSTM0TO → OSTMnTO, OSTMTTOUT → OSTMnTTOUT, OSTMnTT.OSTMnTT → OSTMnTS.OSTMnTS, OSTM0 → OSTMn)
		Section 21 Advanced Timer Unit IV (ATU-IV)	
		All	In the ATU-IV section, noise cancellation mode names were modified as follows: subsequent edge → premature-transition, preceding edge → minimum time-at-level, level-accumulated → level accumulation
		All	In the ATU-IV section, the unit of filtering were modified as follows: individual subblocks/channels → each subblock/channel, individual channels → each channel, individual subblocks → each subblock
		All	In the ATU-IV section, the description was modified as follows: conflict → contention
		1342	Table 21.6 Settings That Determine Noise Cancellation Modes for Timer C, Note 1. modified (x = 0 to 15 → x = 0 to 7)
		1358	Table 21.15 NCMCR2A Register Contents functional description on bit5 to bit0 modified
		1365	21.4.2.8 TIOR2A — Timer I/O Control Register 2A , (3) NCEA0 to 6 — Noise Canceler Enable A0 to 6, body modified
		1376	21.4.2.15 NCNTAx — Noise Canceler Counters Ax (x = 0 to 6), body modified
		1379, 1380	21.4.3.1 Operation of Noise Canceler, body modified
		1379	Figure 21.7 Example of Noise Cancellation in Premature-Transition Cancellation Mode, title of the figure modified
		1380	Figure 21.8 Example of Noise Cancellation in Premature-Transition Cancellation Mode for Two Types of Input Waveforms, title of the figure modified
		1381	Figure 21.9 Example of Noise Cancellation in Minimum Time-at-Level Cancellation mode, title of the figure modified
		1382	21.4.3.2 Operation of Free-Running Counter, body modified
		1388	Figure 21.15 Timer B Block Diagram, description in the figure modified
		1389, 1390	21.5.2.1 TCRB — Timer Control Register B, body modified
		1392	21.5.2.2 TIORB — Timer I/O Control Register B , (6) IOB6 — I/O Control B6, body modified
		1393	Table 21.29 TSRB Register Contents, functional description of bits 9, 6, 4 to 2, and 0 modified
		1394 to 1396	21.5.2.3 TSRB — Timer Status Register B, body modified
		1407	21.5.2.12 TCNTB1 — Event Counter B1, body modified

Rev.	Date	Description	
		Page	Summary
1.20	May 31, 2017	1408	21.5.2.13 OCRB1 — Output Compare Register B1, body modified
		1409	21.5.2.14 OCRB10 — Output Compare Register B10, body modified
		1413	21.5.2.18 ICRB2 — Input Capture Register B2, body modified
		1416	21.5.2.21 TCNTB2 — Reload Counter B2, body modified
		1417	21.5.2.22 PIMR1 — Pulse Interval Multiplier Register 1, body modified
		1419	21.5.2.24 TCNTB6 — Multiplied Clock Counter B6, body modified
		1425	21.5.2.30 TCNTB3 — Correcting Event Counter B3, body modified
		1427	21.5.2.32 TCNTB4 — Multiplied-and-Corrected Clock Counter B4, body modified
		1447	Figure 21.27 Operation of TCNTB5 (with Correction at End of Cycle), erroneous values of TCNTB3 modified: 000040 00060 00000 00020 → 00060 00080 00000 00020
		1448	Figure 21.28 Operation of TCNTB5 (without Correction at End of Cycle), erroneous values of TCNTB3 modified: 000040 00060 00000 00020 → 00060 00080 00000 00020
		1452	Figure 21.31 Block Diagram of Timer C, Note added to "Sub Block:C5", "Sub Block:C6", and "Sub Block:C7", and Note modified
		1463	21.6.2.4 TSRCx — Timer Status Registers Cx (x = 0 to 7), (1) OCMFCxy — Output Compare Match Flag Cxy, title modified
		1464	21.6.2.4 TSRCx — Timer Status Registers Cx (x = 0 to 7), (3) IMFCxy — Input Capture/Compare Match Flag Cxy, body modified
		1470	21.6.2.7 TCNTCx — Timer Counters Cx (x = 0 to 7), body modified: TCNTCn → TCNTCx
		1471, 1472	21.6.2.8 GRCxy — General Registers Cxy, body modified, description added
		1476	21.6.2.11 OCRCxy — Output Compare Registers Cxy, description added
		1478	21.6.2.13 CUCRCx — Counter Upper-Limit Setting Compare Registers Cx, description added
		1488	21.6.3.2 Compare Match Mode — Compare Match Function, body modified
		1489	21.6.3.3 PWM Function, body modified: bit PWMx → bit PWMx0
		1492	Figure 21.37 Operation when the Compare Register Is Modified during One-Shot Pulse Output, omission of the vertical line modified
		1496	Figure 21.42 One-Shot Pulse Output Cancellation Timing Chart, omission of the vertical line modified
		1500	21.7.1 Operation Overview, description of DMAC modified
		1501	Figure 21.45 Block Diagram of Timer D modified ("x = 0 to 9 (number of subblocks), y = 0 to 3 (number of channels)" deleted)
		1505	21.7.2.2 TCRDx — Timer Control Registers Dx, (1) OBREDx — Timer Offset Base Register Enable, body modified
		1506	21.7.2.2 TCRDx — Timer Control Registers Dx, (4) CLR2Dx — TCNT2Dx Clear, title and body modified
		1506	21.7.2.2 TCRDx — Timer Control Registers Dx, (6) CLR1Dx — TCNT1Dx Clear, title and body modified
		1510, 1511	21.7.2.4 TIOR2Dx — Timer I/O Control Register 2Dx, body modified
		1516	Table 21.89 DSR2Dx Register Contents, functional description of bits 7 to 0 modified
		1519	21.7.2.11 DCRDx — Down Counter Control Registers Dx, (1) TRGSELDxy — Down Counter Start/Stop Trigger Select Bits Dxy, body modified
		1522	21.7.2.12, (4) CMFADxy — Compare Match A Flag Dxy, functional description modified (OCRDxy → ODR1Dxy)
		1523	21.7.2.12, (5) CMFBDxy — Compare Match B Flag Dxy, description modified: output compare B register (OCR2Dxy) → output compare register 2Dxy (OCR2Dxy)
		1528	21.7.2.16 TICTSELDx — Timer Input Capture Trigger Select Register Dx, Note 1 modified
		1533	21.7.2.21 OCR1Dxy — Output Compare Registers 1Dxy, description added
		1534	21.7.2.22 RCR1Dx — Range Comparison Value Setting Register 1Dx, body modified (OCR1Dxy + RCR1Dx -> (OCR1Dxy + the value of the range specified by RCR1Dx))
		1536	21.7.2.23 OCR2Dxy — Output Compare Registers 2Dxy, description added
		1537	21.7.2.24 RCR2Dx — Range Comparison Value Setting Register 2Dx, body modified (OCR2Dxy + RCR2Dx -> (OCR2Dxy + the value of the range specified by RCR2Dx))
		1543, 1547	21.7.3 Operation, body modified

Rev.	Date	Description	
		Page	Summary
1.20	May 31, 2017	1548	21.7.3.1 Range Comparison Function, body modified (OCR1Dxy + RCR1Dx -> (OCR1Dxy + the value of the range specified by RCR1Dx))
		1548	Figure 21.49 Operation of the Range Comparison Function, description in the figure modified ("OCR1Dxy + RCR1D" → "OCR1Dxy + the range value specified by RCR1Dx")
		1576	Figure 21.55 Operation of PWM (1) modified ("omice0" deleted)
		1582	21.9.2.1 TSTRF — Timer Start Register F, body modified: 0000 0000_H→ 0000_H
		1588	21.9.2.4 NCCRF — Noise Canceller Control Register F, body modified: 0000 0000_H→ 0000_H
		1598	21.9.2.10 TSRFx — Timer Status Registers Fx, body modified
		1610	21.9.2.20 GRDFx — General Registers DFx, Note 1 deleted, value after reset of bits 31 to 16 in a bit chart modified
		1639	Table 21.142 TSRGx Register Contents, functional description of OVFGx bit modified
		1639	21.10.2.3 TSRGx — Timer Status Registers Gx, body modified
		1647	21.11.2.2 TSRH — Timer Status Register H, body modified
		1660	21.12.2.4 TSRJx — Timer Status Register Jx, body modified
		1682	Figure 21.77 Data DMA Trigger Selection Circuit, the path of the DFE filter completion signal has been corrected to show that it passes through the synchronization circuit.
		1684	Figure 21.79 SAR-AD Trigger Selection Circuit, signal name modified (Timer G1 compare interrupt→Timer G2 compare interrupt)
		1686	21.14.1.2 Contention between Writing to Counter and Input Capture, bit name in the table modified (FDOVJx → FDOVFJx)
		1687	21.14.1.3 Contention between Setting and Clearing of Input Capture Status Flag, bit name in the table modified (FDOVJx → FDOVFJx)
		1689	21.14.2.2 Contention between Writing to CYLRExy and Cycle Match of TCNTExy, entry in the table modified (OCRB6 → OCRB6/OCRB7)
		1692	21.14.2.6 Contention between Writing to TCNTExy and Counter Clearing by Cycle Match, entry in the table modified (OCRB6 → OCRB6/OCRB7)
		1697	21.14.3.1 Contention between Data Transfer and Writing to Transfer Destination Register, "Transfer Data" modified (LDB + PIMR→LDB - PIMR)
		1698	21.14.3.2 Contention between Data Transfer and Writing to Transfer Source Register,"Transferred Value" modified (LDB + PIMR→LDB - PIMR)
		1702	Figure 21.98 Contention between TCNT1H Counter Overflow and Compare Match, erroneous values of TCNT1H modified
		1704	21.14.5.1 Contention between Writing to Noise Canceler Counter and Compare Match with Noise Canceler Register, body modified
		1705	21.14.5.2 Contention between Writing to Noise Canceler Register and Compare Match with Noise Canceler Counter, body modified
		1712	21.14.7.5 Contention between Clearing of the Counter to 0 by Timer B and Clearing of the Counter by the Counter Upper-Limit Setting Function, added
		1713, 1714	21.14.8 Different Specifications of Operation in Response to a Match in Comparison, added
		Section 22 Autonomous Pulse Adapter (APA)	
		1741	22.4.2.2 APAA0CHEN — APAA0 Channel Operation Enable Register, bit chart modified
		1743	22.4.2.3 APAA0CHST — APAA0 Channel Output Status Register, bit chart modified
		1753	22.4.4.2 APAA0EVSW — APAA0 Software Event Register, bit chart modified
		1756	22.4.4.4 APAA0ESTA — APAA0 Event Status Register A, bit chart modified
		Section 23 Motor Control Timer (TSG2)	
		All	Superfluous leading zeros in indices (m and k) have been deleted to alter the notation from two digits to a single digit (except those from 10 to 12).
		1850	Table 23.6 TSG2nCTL0 Register Contents, functional description of TSG2nDWD bit modified (reference modified)
		1853	23.4.2.4 TSG2nCTL4 — TSG2n Control Register 4, bit name of bits 4 to 0 in a bit chart modified (TSG2nRCC[0:0]4 → TSG2nRCC[04:00])
		1853	Table 23.9 TSG2nCTL4 Register Contents, bit name of bits 4 to 0 modified (TSG2nRCC[4:0] → TSG2nRCC[04:00])

Rev.	Date	Description	
		Page	Summary
1.20	May 31, 2017	1855, 1857, 1905, 1932, 1961	Bit name of bits 4 to 0 in TSG2nCTL4 register modified (TSG2nRCC4 to TSG2nRCC0 →TSG2nRCC04 to TSG2nRCC00)
		1864	Table 23.18 TSG2nSTR2 Register Contents (1/2), functional description of bits 9 to 5 modified
		1865	Table 23.18 TSG2nSTR2 Register Contents (2/2), functional description of bits 3 and 2 modified
		1875	23.4.2.22 TSG2nCMP1W — TSG2n Compare Registers 1 and 2, entry under "PWM Mode" of TSG2nCMP2 modified
		1876	23.4.2.23 TSG2nCMP5W — TSG2n Compare Registers 5 and 6, entry under "120-DC Mode" of TSG2nCMP6 modified
		1884	23.4.2.31 TSG2nPAT0W — TSG2n Pattern Register 0, body modified (bits 2 and 1 → bits 2 to 0)
		1885	23.4.2.32 TSG2nPAT1W — TSG2n Pattern Register 1, body modified (bits 2 and 1 → bits 2 to 0)
		1895	23.5.1 Basic Operation of 16-Bit Counter, description of INTTSG2nI[7] modified (TSG2nCMP6→TSG2nCMP7)
		1899	23.5.3 Compare Register Rewrite Operation, Anytime rewriting, body modified ("Caution" deleted)
		1900	Table 23.31 Active Width Setting in SP-PWM Mode, register name modified
		1901	Figure 23.8 Anytime Rewriting Timing (Example in PWM Mode), modified
		1903	23.5.3.2 Operation Example of Reloading (Simultaneous Rewriting), body modified (TSG2nCTL3 → TSG2nCTL4)
		1904	Figure 23.11 Simultaneous Rewriting Timing (2/2) (Example of PWM mode), interrupt modified (INTTSG2nIPEK → INTTSG2nI0)
		1906	23.5.4.1 Timer Output in Each Mode, body modified (timer outputs (TSOn1-TSOn6 and TSOn7)) → timer outputs (TSOn0 to TSOn7))
		1909	Table 23.39 List of Interrupts in Each Mode (3/5), entries under "INTTSG2nI[11]" modified
		1916	Figure 23.18 TSOn0 Pin Output depending onTSG2nIOC1.TSG2nTOS Setting, bit name modified (TSG2nCUF → TSG2nSUF)
		1917	Figure 23.19 Example of Positive /Negative Phase Simultaneous Activation Detection Flag Operation, bit name and description modified
		1919	Figure 23.22 Example of Noise Detection Flag Operation, bit name and description modified
		1920	Figure 23.23 Example of Pattern Order Detection Flag Operation (Normal Operation), bit name modified (TSG2nMDF → TSG2nTSF)
		1921	Figure 23.25 Example of Operation when Values Input to Two Pins of TAPTSn2-TAPTSn0 Change (Abnormal Operation), bit name modified (TSG2nTBF → TSG2nTSF)
		1922	Figure 23.26 Example of Pattern Error Detection Flag Operation (TAPTSn2-TAPTSn0 Pins = 111), bit name and description modified
		1932	Figure 23.35 When TSG2nRMC = 0 and TSG2nRIA = 0 in TSG2nCTL3 (without Reload Skipping), bit name modified
		1937	Figure 23.38 When TSG2nPIE = 1, TSG2nVIE = 1, and TSG2nRCC04 to TSG2nRCC00 = 00_B in TSG2nCTL4, and TSG2nACC01 and TSG2nACC00 = 00_B in TSG2nCTL5 (HT-PWM Mode), names of interrupts and values of bits modified
		1938	Figure 23.39 When TSG2nPIE = 0, TSG2nVIE = 1, and TSG2nRCC04 to TSG2nRCC00 = 02_B in TSG2nCTL4 and TSG2nACC01 and TSG2nACC00 = 00_B in TSG2nCTL5 (HT-PWM Mode), bit name modified (TSG2nICE → TSG2nPIE)
		1940	Figure 23.42 Example of Operation of A/D Conversion Trigger Skipping Function, description modified
		1943	Figure 23.45 Example of Error Interrupt Operation, bit name and description modified
		1971	23.11.2.8 Notes Concerning Dead Time Control in HT-PWM Mode, CAUTIONS 3 and 4 modified
		1974	Figure 23.64 Flow of Software Output Control in HT-PWM Mode, description in step (2) modified
		1980	Table 23.59 Compare Registers and Dead Time Setting Register Functions in SP-PWM Mode, a row for TSG2nDCMP0W and TSG2nDCMP2 registers added
		1995, 1996	23.11.4.3 Control Methods in 120-DC Mode, body modified
		2003, 2004	23.11.4.6 List of Output Patterns in 120-DC Mode, description added

Rev.	Date	Description	
		Page	Summary
1.20	May 31, 2017	2006	Figure 23.79 Control when Timer Output Starts in Reverse Rotation (when Normal Pattern is Input), pattern name modified
		2006	Figure 23.80 Control when Timer Output Starts in Normal Rotation (when Error Pattern is Input), modified
		2007	Figure 23.81 Control when Timer Output Starts in Reverse Rotation (when Error Pattern is Input), names of the pattern modified
		2018	Figure 23.91 Example of Switching from 120-DC Mode to Software Output Control Function, TSG2nSTCI0 signal deleted
		Section 24 Timer Option (TAPA)	
		2024	Table 24.3 TAPAnCTL0 Register Contents, functional description of TAPAnDCM bit modified (TAPAnOPHS0 deleted)
		2026	Table 24.6 TAPAnACTS Register Contents, functional description of TAPAnACTS bit modified (TAPAnACE → TAPAnACWE)
		2027	Table 24.7 TAPAnACTT Register Contents, functional description of TAPAnACTT bit modified (TAPAnACE=0 → TAPAnACWE=1)
		2031	24.3.2.3 Operating Procedure Example for Hi-Z Control in Response to Asynchronous Input, description in the table modified ((if TAPAnDCM = 0) and (if TAPAnDCM = 1) deleted, TAPAnASTT bit → TAPAnACTT bit)
		Section 25 Peripheral Interconnection (PIC)	
		All	Terms unified, channel group → scan group
		2034	Table 25.5 PIC1SSER2 Register Contents, functional description of bits 9 and 8 modified
		2035	Table 25.6 PIC1SSER3 Register Contents, functional description of bits 2 to 0 modified
		2041	Table 25.10 List of PIC2 Registers (1/3), register name modified (A/D converter trigger edge selection control register→A/D converter 1 trigger edge selection control register)
		2068	25.2.4.1 ADC Trigger Selection Function, body modified (ADCA hardware triggers→ADCB hardware triggers)
		Section 26 A/D Converter (ADCB)	
		2129	26.7.9.2 A/D Converter Self-Diagnosis, [Settings], description modified ((n = 0 to 47) deleted), and NOTE added
		2131	26.7.9.3 Wiring-Break Detection Self-Diagnosis, [Settings], description modified ((n = 0 to 39) deleted), and NOTE added
		2132	Figure 26.18 Timing Chart of Normal A/D Conversion Operation (Single Conversion), description in the figure modified (tCONV → tSAR, tSCAN → tSG)
		2137	Figure 26.22 Scan Conversion End Interrupt Occurrence Timing, peripheral clock modified
		2138	Figure 26.23 Example of an MPX Interrupt Occurrence, peripheral clock modified
		2139	Figure 26.24 Example of an Occurrence of A/D Error Interrupt and A/D Parity Error Notification, peripheral clock modified
		2144	26.9.3 Notes when Current is Being Injected, description in (2) modified (GCTRL bits → bits 4 and 3 (GCTRL))
		2155	26.11.5 Operation, [Setting Procedure], description in step 4 modified (from the first time → from the second time)
		Section 27 Delta-Sigma AD Converter (DS-ADC)	
		2177	27.5.5 DSADCmADCR — AD Control Register (m = 0 to 7), CAUTION 2 modified
		2178	27.5.6 DSADCmADSR — AD Status Register (m = 0 to 7), CAUTION modified
		2183	27.5.11 DSADCmFCR — Digital Filter FIR Control Register (m = 0 to 7), CAUTION modified (DSADCCOSMPRCRm → DSADCCOSMPRCR)
		2186	27.6.2 Setting Filter Type, body modified (DSADCCOSMPRCRm → DSADCCOSMPRCR)
		2186	Table 27.22 A/D Conversion Processing Time (Unit: Pϕ P-Bus Clock (40 MHz)), entries in the table modified and Note 1 added
		2187	Figure 27.4 A/D Conversion Start and End, Notes 1 and 2 added
		2188	27.6.4 Starting A/D Conversion by an External Trigger, description added
		2188	27.6.5 Terminating A/D Conversion by an External Trigger, description modified

Rev.	Date	Description	
		Page	Summary
1.20	May 31, 2017	Section 28 Digital Filter (DFE)	
		2197	Table 28.2 List of Terms (1/2), PH index register deleted
		2214	Table 28.11 TRGCHn Register Contents (1/2), functional description of PFE and AFE bits modified
		2223, 2224	Table 28.19 PITRG Register Contents, functional description modified
		2225, 2226	Table 28.20 MITRG Register Contents, functional description modified
		2227, 2228	Table 28.21 FITRG Register Contents, functional description modified
		2234	Figure 28.10 Usage of Memory, description in the figure modified
		2239	28.4.7.1 Timer Trigger Input, body modified
		2239	Figure 28.13 Timer Trigger Input, description in the figure modified
		2242	Table 28.28 Trigger Flag Functions and Processing, "Output data write" column of "PH performed" for "Accumulation/decimation disabled", modified (Not provided→Provided)
		2272	28.5.10 Restrictions on Trigger Setting Registers, body modified
		Section 29 Safety	
		2284	29.2.2.3 Details of Registers, (6) CF1STERSTR_VCI/_PE1/_PCU — Code Flash 1st Error Status Register, description added
		2288	29.2.2.4 Test Function, (4) Self-Diagnosis of the ECC and Address Parity for Code Flash Memory, added
		2303	Table 29.32 LRTDATBFn_PE1 Register Contents, functional description of LRTDATBF bits modified
		2308	29.2.4.3 Details of Registers, (8) LR1STERSTR_PE1 — Local RAM 1st Error Status Register, description added
		2310	29.2.4.4 Test Function, (6) Self-Diagnosis of the Address Parity Check Function, description in (b) modified
		2314	29.2.5.3 Details of Registers, (2) LRFSTERSTR_PCU — Local RAM 1st Error Status Register, description added
		2323	Figure 29.2 ECC of Global RAM and Address Parity, location of the figure modified
		2335	29.2.6.3 Details of Registers, (10) GR1STERSTR_VCI/PE1 — Global RAM 1st Error Status Register, description added
		2359	29.2.9.3, (1) E710CTL — ECC Control Register, R/W of bit 8 in a bit chart modified to R
		2372	29.2.10.3, (1) E610CTL — ECC Control Register, R/W of bit 8 in a bit chart modified to R
		2418	29.5.4.3 MISR1L_PE1/PCU — Multi-Input Signature Register 1L, body modified
		2419	29.5.4.4 MISR1H_PE1/PCU — Multi-Input Signature Register 1H, body modified
		2420	29.5.4.5 MISR2L_PE1/PCU — Multi-Input Signature Register 2L, body modified
		2421	29.5.4.6 MISR2H_PE1/PCU — Multi-Input Signature Register 2H, body modified
		2437	Table 29.142 CLMATEST Register Contents, functional description of MONCLKMSK bit modified
		2439	29.6.4 Detection of Abnormal Clock Frequency, description modified, and Note 1 added
		2442	29.6.5 Self-Diagnosis, (1) and (4) added, and description in (3) and (5) modified
		Section 30 Error Control Module (ECM)	
		2448	Table 30.2 List of Error Sources and Safety Processing (2/2), Note 11 added
		2453	Figure 30.1 Outline of ECM, description in the figure modified (ERRORUT output → ERROROUT output)
		2459	Table 30.11 ECMmESSTR1 Register Contents, functional description of bit 31 modified
		Section 31 Data CRC (DCRA)	
		2489	Table 31.4 DCRA0COUT Register Contents, functional description of DCRA0COUT[31:0] bits modified (DCRAnCOUT → DCRA0COUT)
		2489	31.2.3 DCRA0COUT — CRC Data Register, CAUTION modified (DCRAnCIN register → DCRA0CIN register)
		2490	31.2.4 DCRA0CTL — CRC Control Register, CAUTION 2 modified (DCRA0CTL.DCRA0ISZn[1:0] → DCRA0CTL.DCRA0ISZ[1:0], DCRAnCOUT register → DCRA0COUT register)

REVISION HISTORY	RH850/E1M-S User's Manual: Hardware

Rev.	Date	Description	
		Page	Summary
1.20	May 31, 2017	Section 33 Flash Memory	
		2533	33.11 Usage Notes, (7) Items prohibited during programming, erasure, and blank checking, description modified (programming and erasure → programming, erasure, and blank checking)
		Section 35 RAM	
		2543	Table 35.2 TM_CC Register Contents, description of bits 31 to 1 added
		2547	Table 35.6 TM_MA0 Register Contents, CAUTION in the functional description modified
		2547	35.4.9 TM_MA0 — Tuning Memory Mapping Address Register 0, CAUTION 3 deleted
		2548	35.4.11 Notes on Access to ERAM, description (lockstep compare error) added
		2549	35.5 Usage Notes, description added
		Section 37 Electrical Characteristics	
		2575	Table 37.11 DC Characteristics (Input Capacitance), typical value modified (1.0 → 10)
		2576	37.2.11 Supply Current Characteristics, CAUTION 1 modified (AVREFH pin → A1VREFH pin)
		2578	Figure 37.3 AC Measurement Conditions, IOL*2 added
		2604	Table 37.37 ADC Converter Characteristics, Note 1 modified (AVREFHn → AnVREFH)
		2610	Table 37.44 Thermal Resistance of RH850/E1M-S, a row for 4LΨjt added and Note modified
		2610	Table 37.45 JESD51-9 Compliant Board (4 layers), title of the table and entries in the table modified

RH850/E1M-S User's Manual: Hardware

Publication Date: Rev.1.00 Apr 21, 2015
Rev.1.20 May 31, 2017

Published by: Renesas Electronics Corporation

SALES OFFICES Renesas Electronics Corporation http://www.renesas.com

Refer to "http://www.renesas.com/" for the latest and detailed information.

Renesas Electronics America Inc.
2801 Scott Boulevard Santa Clara, CA 95050-2549, U.S.A.
Tel: +1-408-588-6000, Fax: +1-408-588-6130

Renesas Electronics Canada Limited
9251 Yonge Street, Suite 8309 Richmond Hill, Ontario Canada L4C 9T3
Tel: +1-905-237-2004

Renesas Electronics Europe Limited
Dukes Meadow, Millboard Road, Bourne End, Buckinghamshire, SL8 5FH, U.K
Tel: +44-1628-585-100, Fax: +44-1628-585-900

Renesas Electronics Europe GmbH
Arcadiastrasse 10, 40472 Düsseldorf, Germany
Tel: +49-211-6503-0, Fax: +49-211-6503-1327

Renesas Electronics (China) Co., Ltd.
Room 1709, Quantum Plaza, No.27 ZhiChunLu Haidian District, Beijing 100191, P.R.China
Tel: +86-10-8235-1155, Fax: +86-10-8235-7679

Renesas Electronics (Shanghai) Co., Ltd.
Unit 301, Tower A, Central Towers, 555 Langao Road, Putuo District, Shanghai, P. R. China 200333
Tel: +86-21-2226-0888, Fax: +86-21-2226-0999

Renesas Electronics Hong Kong Limited
Unit 1601-1611, 16/F., Tower 2, Grand Century Place, 193 Prince Edward Road West, Mongkok, Kowloon, Hong Kong
Tel: +852-2265-6688, Fax: +852 2886-9022

Renesas Electronics Taiwan Co., Ltd.
13F, No. 363, Fu Shing North Road, Taipei 10543, Taiwan
Tel: +886-2-8175-9600, Fax: +886 2-8175-9670

Renesas Electronics Singapore Pte. Ltd.
80 Bendemeer Road, Unit #06-02 Hyflux Innovation Centre, Singapore 339949
Tel: +65-6213-0200, Fax: +65-6213-0300

Renesas Electronics Malaysia Sdn.Bhd.
Unit 1207, Block B, Menara Amcorp, Amcorp Trade Centre, No. 18, Jln Persiaran Barat, 46050 Petaling Jaya, Selangor Darul Ehsan, Malaysia
Tel: +60-3-7955-9390, Fax: +60-3-7955-9510

Renesas Electronics India Pvt. Ltd.
No.777C, 100 Feet Road, HAL II Stage, Indiranagar, Bangalore, India
Tel: +91-80-67208700, Fax: +91-80-67208777

Renesas Electronics Korea Co., Ltd.
12F., 234 Teheran-ro, Gangnam-Gu, Seoul, 135-080, Korea
Tel: +82-2-558-3737, Fax: +82-2-558-5141

Colophon 4.1

RH850/E1M-S

Renesas Electronics Corporation

R01UH0466EJ0120